UNITED STATES DISTRICT COURT
FOR THE NORTHERN DISTRICT OF CALIFORNIA
SAN FRANCISCO DIVISION

ORACLE, INC. Plaintiff, v. GOOGLE INC., Defendant.	Case No. CV 10-03561 WHA

HIGHLY CONFIDENTIAL

EXPERT REPORT OF DR. GREGORY K. LEONARD

CORRECTED (MARCH 10, 2016)

Contents

I. Qualifications

1. I am an economist and partner at Edgeworth Economics, 333 Bush Street, Suite 1450, San Francisco, CA 94104. I received a Bachelor of Science degree in Applied Mathematics-Economics from Brown University in 1985 and a Ph.D. in Economics from the Massachusetts Institute of Technology in 1989. Prior to joining Edgeworth Economics, I was, at various times, a senior vice president with NERA Economic Consulting, a senior vice president with Lexecon Inc., a founding member and director of Cambridge Economics, Inc., and an assistant professor of economics at Columbia University.

2. My specialties within economics are applied microeconomics, which is the study of the behavior of consumers and firms, and econometrics, which is the application of statistical methods to economics data. I have published over sixty papers in scholarly and professional journals. My publications are listed on my curriculum vitae, attached as Appendix A. A number of these papers address issues in industrial organization, demand for products, intellectual property and the calculation of damages in patent infringement litigation, and econometrics, including publications in the *Journal of Industrial Economics*, the *RAND Journal of Economics*, the *Journal of Econometrics*, the *Berkeley Journal of Technology and Law*, and *les Nouvelles*.

3. For example, a paper in the *Journal of Econometrics* addresses econometric approaches to estimating patent value for use in patent litigation and licensing negotiations; a recent paper in *Antitrust* focuses on methodologies for determining "reasonable and non-discriminatory royalties" for standard-essential patents; and a paper in the *Columbia Science and Technology Review* discusses the concept of apportionment in patent valuation.

4. I am a senior editor of the *Antitrust Law Journal* and have served as a referee for numerous economics and other professional journals. I have given invited lectures on intellectual property and antitrust issues at the Federal Trade Commission (FTC), the United States Department of Justice (DOJ), the Directorate General for Competition of the European Commission, the Fair Trade Commission of Japan, and China's Supreme People's Court and Ministry of Commerce. I have been retained by the DOJ to consult on antitrust matters.

5. In 2009, I was invited to speak at a session of the FTC's hearings on the "Evolving IP Marketplace" concerning the calculation of patent damages. In the report that the FTC subsequently issued, my views on damages calculation were cited extensively.[1] In 2007, I served as a consultant to and testified before the Antitrust Modernization Commission, which was tasked by Congress and the President of the United States to make recommendations for revising U.S. antitrust laws. In its *Uniloc* decision, the U.S. Court of Appeals for the Federal Circuit cited one of my publications in support of its conclusion that a method of calculating reasonable royalty damages in a patent case (the so-called "25% Rule") is an unreliable and flawed methodology.[2]

6. I have served as an expert witness in a number of litigation matters before U.S. District Courts, the (U.S.) International Trade Commission, state courts, and arbitration panels. A list of cases in which I have testified (in deposition or at trial) in the last four years is provided in my curriculum vitae, attached as Appendix A. My hourly rate for this matter is $800.

---

[1] Federal Trade Commission, *The Evolving IP Marketplace: Aligning Patent Notice and Remedies with Competition*, March 2011.

[2] *Uniloc USA, Inc. v. Microsoft Corp.*, 632 F.3d 1292, 1313(Fed. Cir. 2011).

II. Assignment and Materials Considered

7. I understand that Oracle alleges that the "structure, sequence, and organization" (SSO) and declaring code for 37 specific application programming interface (API) packages in the Android operating system infringe Oracle copyrights.[3] I have been asked by Google to assess the appropriate damages assuming Google does not prevail on its fair use defense or equitable defenses and thus is found liable for copyright infringement.

8. In the course of my analysis, I have reviewed the documents and other information listed in Appendix B to this report. Specific documents and other information cited as support in this report are not meant to be an exhaustive listing of all such documents or information.

9. Regarding my anticipated trial testimony in this action, I may use as exhibits various documents or other materials relevant to the issues addressed in this report. I also reserve the right to use demonstrative exhibits, enlargements of actual exhibits, animations and any other kind of information in order to convey my opinions. I reserve the right to supplement my report, for example, if additional information becomes available.

III. Summary of Opinions

10. I have reached the following opinions.

11. I understand that a plaintiff can seek "unjust enrichment" damages, which are any profits of the infringer that are attributable to the infringement and are not taken into

[3] I discuss the SSO and declaring code interchangeably in this report because I understand that the accused declaring code is the series of naming conventions that, taken collectively, constitutes the accused SSO. Although the two concepts may be discussed separately, Oracle's technical and damages experts analyze the SSO and declaring code as one effective entity.

account in computing the plaintiff's actual damages.[4] I also understand that a plaintiff "must first show a causal nexus between the infringement and the gross revenue"; and, "once the causal nexus is shown, the infringer bears the burden of apportioning the profits that were not the result of infringement."[5] I have concluded that:

- Oracle's expert Mr. Malackowski's causal nexus analysis is fundamentally flawed because (1) he fails to consider the economically appropriate counterfactual, i.e., the one in which Google pursued its best non-infringing course of action, (2) he fails to acknowledge that most of the revenues he points to are only indirectly related to Android, and (3) he does not consider the large number of factors other than the alleged infringement that contributed to Android's success and thus fails to show a causal nexus or link between the revenues and the alleged infringement.

- Assuming that Mr. Malackowski has met Oracle's burden of identifying revenues that are causally linked to the alleged infringement, Mr. Malackowski substantially understates the costs that are associated with the revenues he claims have a causal nexus to the alleged infringement. I have made corrections to Mr. Malackowski's cost calculations. I conclude that the profits on Android-related revenue (both direct and indirect) for the period 2009 to 2015 are [redacted], before accounting for ad revenue recapture on non-Android mobile devices, and [redacted], accounting for such recapture.

- I understand that the Android-related profits must next be apportioned to reflect the element of profit attributable to factors other than the alleged infringement. I understand that the purpose of apportionment is to subtract out the contribution of these other factors, leaving the contribution of the allegedly infringing material to the profits.

- As a matter of economics, apportionment may be performed from the bottom up, or from the top down. The bottom up approach involves directly measuring the profits contributed by the other factors, or the flip side of that, the profits contributed by the allegedly infringing material.

---

[4] 17 U.S.C. § 504(b).

[5] *Polar Bear Prods., Inc. v. Timex Corp.,* 384 F.3d 700, 711 (9th Cir. 2004), *as amended on denial of reh'g and reh'g en banc* (Oct. 25, 2004), *opinion amended on denial of reh'g,* No. 03-35188, 2004 WL 2376507 (Oct. 25, 2004).

The top down approach involves determining an apportionment percentage that represents the contributions of the other factors, or the flip side of that, the percentage that represents the contribution of the alleged infringement. The latter percentage is then applied to the overall profit figure.

- The contribution of the alleged infringement to the Android-related profits was to decrease the costs that Google had to incur in providing for third party applications. Specifically, the contribution of the alleged infringement was that it allowed Google to avoid undertaking one of the following (mutually exclusive) actions: (1) implement the OpenJDK class libraries for the 37 API packages prior to the release of Android, (2) train any third party developers who developed for Android *only* because they were familiar with the Java language, or (3) promote the development of third party applications by developers who developed for Android *only* because they were familiar with the Java language. The bottom up approach to apportionment involves measuring the cost-savings associated with avoiding each of these actions. I have calculated these cost-savings to be $85,000, $1.9 million, and $23 million to $100 million, respectively. Unjust enrichment damages should be no more than the minimum of these three figures, or $85,000. Even in the absence of these actions, I have concluded that the contribution of the alleged infringement to the number of Android users (and thus Android-related profits) was small, limited to at most $203 million of the Android-related profits.

- I also have implemented two top down approaches to apportionment. In the first approach, the first step is to apportion the Android-related profits between Android, on the one hand, and Google's search and ad technologies and services, on the other. I have concluded that [redacted] of the Android-related profits should be apportioned to Android (with the remainder apportioned to Google's search and ad technologies and services). The second step is to apportion [redacted] between the SSO and declaring code of the 37 API packages at issue, on the one hand, and the rest of Android, on the other. Because I have seen no evidence that the 37 API packages required greater programming effort or ingenuity than the rest of Android, lines of code provide a reasonable basis to apportion between the 37 API packages and the rest of Android. The code in the 37 API packages make up about 1.7% of all Android code. Applying this figure to the [redacted] figure yields an unjust enrichment damages figure of [redacted]. This figure is conservative because only the SSO and declaring code is at issue in this case, not the implementing code, yet I have included both in the numerator of the percentage.

- In the second top down approach, I apportion between the SSO and declaring code of the 37 APIs, on the one hand, and the remainder of Android and Google's search and ad technologies and services in a single step, based on lines of code in the 37 API packages divided by a conservative estimate of the lines of code in Android and Google's search and ad code bases. This percentage, 0.4%, is then applied to the [redacted] in Android-related profits to obtain an unjust enrichment damages figure of [redacted].

12. I also understand that a plaintiff can seek its actual damages, such as lost profits, subject to there being no double-counting between any unjust enrichment damages and any actual damages. Regarding actual damages, I have concluded:

- Mr. Malackowski has substantially overstated Oracle's lost Java ME licensing profits. First, he ignores Google's non-infringing alternatives in the but-for world, which would have allowed Google to achieve essentially the same market outcomes as actually occurred. I conclude that the lost Java ME licensing profits due to the alleged infringement are essentially zero. Mr. Malackowski also errs by using a 2008 forecast as his basis for the but-for world. However, such a forecast by definition does not incorporate any of the events that were unanticipated in 2008, but affected Java ME licensing revenues. Such events would include, for example, the success of the iPhone, the adverse consequences of stagnation in the Java ME platform, and the financial crisis of 2008-2009. Correcting Mr. Malackowski's but-for world (while still assuming Android would not have existed, which is incorrect) substantially reduces his calculation of lost Java ME licensing profits.

IV. Legal Framework for Copyright Damages

13. I understand that the legal framework for copyright damages allows a plaintiff to recover actual damages, which can be calculated based on, for example, the plaintiff's lost profits that are caused by the alleged copyright infringement. I also understand that a plaintiff can recover the infringer's profits that are attributable to the alleged copyright infringement if these profits are not already accounted for in computing actual damages or statutory

damages.[6] A calculation based on the profits of the infringer, sometimes referred to as unjust enrichment or disgorgement, first requires the plaintiff to identify revenues earned by the alleged infringer having a causal nexus to the alleged infringement. I understand that, once the plaintiff carries its burden of proving the amount of the alleged infringer's revenues causally linked to the alleged infringement, it is then the burden of the alleged infringer "to prove his or her deductible expenses and the elements of profit attributable to factors other than the copyrighted work."[7]

V. Revenues and Costs Indirectly Associated With Android

A. Android Does Not Generate Any Revenues Directly

14. An important initial point is that Android does not generate any revenues for Google directly. That is, no one pays Google for the use of the Android operating system. That software is available for free to anyone who wishes to use it under the terms of an open-source license offered by Google.

15. Google does earn revenue indirectly from Android. First, Google earns revenue from Android through direct sales of hardware, including smartphones and tablets, that run the Android operating system. Second, and less directly related to the Android operating system software, Google earns revenue from its Google Play store, which offers for sale applications and digital content that Android users can download and/or play on their Android devices. (Some content available on Google Play, such as Google Play Music, may also be played on desktop or laptop computers that do not use the Android operating system, so not all the

[6] 17 U.S.C. § 504.

[7] Id.; *Polar Bear,* 384 F. 3d at 711.

revenue associated with digital content sold by Google Play is associated with the Android operating system.) Third, and most removed, Google earns advertising revenues when, for example, Android users perform Google searches on Android devices. In his opening damages report in this case, Oracle's expert Mr. Malackowski claims that these revenue streams have a causal nexus to the infringement alleged in this case.

16. With respect to the third category identified above—advertising revenue earned by Google from Google searches performed on Android devices—the relationship between the revenue earned by Google and the Android operating system is particularly attenuated, because Google also serves advertisements on other devices, such as non-Android mobile devices (e.g., iPhones and iPads using Apple's iOS operating system and Blackberry and Microsoft devices using those companies' respective proprietary operating systems) and personal computers. These channels of distribution are substitutes on both the supply side (i.e., they are alternative means for Google and advertisers to reach users) and the demand side (i.e., they are alternative means for a user to perform a Google search).

17. Another reason why advertising revenue on Android devices is particularly far removed from the content of the Android operating system is the fact that Google's search and advertising technologies and services had already been developed and were well-established on other platforms (e.g., personal computers) long before Android launched.[8] For users of Android devices, Google's reputation as the most popular Web-based search engine preceded and existed independently of the Android platform and operating system. With respect to

[8] Google Inc. 10-K, 2007 Annual Report, December 31, 2007, p. 44; Google Inc. 10-K, 2006 Annual Report, December 31, 2006, p. 42; Google Inc. 10-K, 2005 Annual Report, December 31, 2005, p. 48.

advertisers, Google's reputation for effectively delivering targeted advertisements to a desired audience likewise preceded and existed independently of Android.

B. Causal Nexus

18. I understand that Oracle must establish the existence of a causal nexus between the alleged infringement at issue—Google's use of the SSO and declaring code from 37 API packages within the Android operating system—and the revenues and profits it claims Google has earned due to the alleged infringement.

19. Speaking as an economist, the appropriate conceptual way to measure the causal effect of a factor on an outcome variable is to compare the difference in the outcome variable between the actual world and the counterfactual where the factor in question is altered exogenously from its actual value and the rest of the system is allowed to adjust.[9] In the case of an economic system, this means that the economic actors are allowed to re-optimize and choose new actions in the counterfactual.

20. In the context of this case, assessing whether there is a causal nexus between Google's use of the SSO and declaring code of the 37 API packages ("the allegedly infringing material") and a particular revenue stream first requires an analysis of Google's best course of action had it not used the allegedly infringing material. Then, the counterfactual revenue stream and profits that Google would have earned taking its best course of action can be analyzed to determine the extent of the causal nexus, if any.

21. Oracle's expert Mr. Malackowski appears to agree that the causal nexus analysis requires the identification of a counterfactual, because he asserts in his report, and bases his

[9] See, e.g., J. Pearl, Causality: Models, Reasoning, and Inference (2nd Ed., 2009), p. 184.

analysis on the conclusion, that Google had no non-infringing alternatives to using the allegedly infringing materials.[10] However, Mr. Malackowski is incorrect about the counterfactual and, in particular, about the absence of non-infringing alternatives. As discussed in detail elsewhere in this report, Google had a number of viable non-infringing alternatives to using the allegedly infringing material. Because he has identified the wrong counterfactual, Mr. Malackowski's analysis of the causal nexus between the allegedly infringing material and Google's revenues is necessarily flawed. Accordingly, in my opinion and from an economic perspective, Mr. Malackowski has failed to establish the existence of a causal nexus between the alleged infringement and the revenues he identifies.

22. In addition, given that the advertising revenues associated indirectly with Android were the result of many factors having nothing to do with the alleged infringement, such as Google's investments in its search and advertising technologies and services, other functionalities of Android, the investments of device manufacturers in providing high quality Android devices, etc., and given that these advertising revenues would have been captured by Google at least to some extent through other channels (such as iPhone and personal computers) even if Android did not exist at all (which, as will be explained below, is not the correct counterfactual in the event Google did not use the allegedly infringing material), further doubts exist regarding the presence of any causal nexus between the advertising revenues and the alleged infringement.[11] Mr. Malackowski makes no effort to isolate the causal effect, if any,

[10] Malackowski 1/8/2016 Report, ¶¶ 232-238.

[11] These same factors also play a role in apportionment. I understand that the profits for which a causal nexus exists must still be apportioned to the alleged infringement. That is, the portion of the profits corresponding to the contribution of other factors should not be attributed to the alleged infringement.

of the alleged infringement from the causal effects of these other factors on Google's advertising revenue earned on Android devices, and thus has no reasonable basis to conclude that any causal nexus exists as to this category of Google revenue. Instead, Mr. Malackowski applies the same general causal nexus theory to Google advertising revenue that he applied to other revenue that he claims is Android-related.

23. Mr. Malackowski's causal nexus argument can be summarized as follows: if the allegedly infringing material were removed from Android (and Google was not allowed within this counterfactual to adjust Android in any way), Android would not work, and Google would not earn any of the associated revenues and profits. Mr. Malackowski supports his causal nexus argument by asserting that Google's use of the allegedly infringing material gave Google access to existing Java developers, allowed for faster programming, and got Android to market more quickly than otherwise would have been possible. On its face, Mr. Malackowski's causal nexus argument is incorrect because it ignores the fact that, if there had been no Android at all, or if Android had been less attractive to developers resulting in fewer apps being available, the individuals who used Android devices in the real world would have turned in the but-for world to other mobile platforms, for example iOS, in large numbers and Google would earn revenues and profits on ads served to those users' devices. But, Mr. Malackowski's argument is also incorrect because it relies on a simplistic (and misguided) view of but-for causation that is in no way unique or particular to the allegedly infringing work in this case. Mr. Malackowski has not shown that the same but-for causation argument would not apply to myriad blocks of code, or even individual lines of code, within Android—if they were pulled out (and Google was not allowed within the counterfactual to do anything to replace them), Android would no longer

work. For example, can Mr. Malackowski deny that if one were to remove the open-source Linux kernel or the Google-developed Android Runtime from Android, the operating system would not work? Or that if one were to remove from Android any of the non-accused Android APIs or API packages developed by Google or by third parties other than Oracle, Android would not work? This renders Mr. Malackowski's causal nexus argument superficial and meaningless as a matter of basic logic. Finally, as noted above, Mr. Malackowski's analysis is incorrect because his counterfactual is incorrect: to determine how much (if any) value the allegedly infringing material contributed to Google's profits, in evaluating the counterfactual, Google must be allowed to respond optimally to the inability to use the allegedly infringing material, and the proper analysis must take account of the effect (if any) of Google's optimal responses on Google's revenues and profits.

C. Nature of Google's Accounting Practices

24. The P&L statement for Android was compiled for the period from January 2008 through December 2015. The most recent Android P&L documents covering the period from 2011 to 2015 do not include Android Ads revenues, because, at some point during the relevant period, Google's Android business unit made the decision to include on the Android P&L only those revenues and costs earned from sales of hardware and Google Play applications and content, and to exclude advertising revenues and costs, which, as discussed above, are highly attenuated from Android in terms of a causal nexus.[12] Nevertheless, Google maintains data regarding revenues and costs associated with advertising served on Android devices, and, as

[12] Gold 12/11/2015 Dep., p. 64:13-23.

described below, in my calculations I have included Android Ads revenues with the overall Android P&L model.

25. In addition, between the 2008-2010 P&Ls and the 2011-2015 P&L, the cost of sales categories were changed and instead organized according to respective business lines (i.e. Apps, Digital Content, and Hardware) as opposed to cost categories including TAC (when Google included Android Ads directly on the Android P&L in 2008 to 2010), CAC, Data Center, Network, SRE, and Other Cost of Sales.[13]

D. Revenues

1. Hardware Revenues Generated from Android Device Sales

26. Hardware revenues generated from Android device sales are included in the Android P&L.[14] Hardware revenues includes direct to consumer sales of hardware devices such as phones, tablets, watches, and accessories.[15] For instance, the Nexus One phone was released and sold during 2010 and subsequently Google-sold Nexus devices were released in 2012.[16] Google does not manufacture the hardware that is reflected in the Android hardware revenue line; the hardware is instead manufactured by third-party OEMS.[17]

[13] GOOGLE-00395614; GOOG-00103813.

[14] Google provides financial information from 2008 Q1 – 2015 Q4, of which 2015 Q4 is a forecast. GOOG-00103813.

[15] Google has at times sold Nexus and other phones. Gold 12/11/2015 Dep., pp. 33, 70.

[16] "From Nexus One to Nexus 10; A Brief History of Google's Flagship Devices," ARS Technica, May 15, 2013, http://arstechnica.com/gadgets/2013/05/from-the-nexus-one-to-the-nexus-10-a-brief-history-of-nexus-devices/.

[17] Gold 12/11/2015 Dep., p. 75:17-21.

2. Apps Revenues Generated from Sales of Android Apps

27. The Android P&L also includes revenues generated from the sale of applications sold on the Google Play store for use on Android devices.[18] This revenue category includes the revenue that Google records from apps sold as well as in-app purchases made through the Google Play store.[19]

3. Digital Content Revenues

28. In addition, the Android P&L includes revenues from the sale of digital content.[20] Digital content is the revenue recorded from sales of digital content sold through the Google Play store. Digital content includes items such as movies, books, music, TV shows, and magazines, which can be played by the purchaser on Android devices, but also on other devices that do not use the Android operating system, including desktop and laptop computers.[21]

4. Advertising Revenues Generated on Android Devices

29. The "Android revenues" category includes revenues from Ads, Hardware, Apps, and Digital Content. For the 2012 to 2015 period, Ads revenue is composed of Search, AdSense, and Display revenue.[22] Search revenue reflects revenue generated when an Android device user clicks on an ad appearing in Google search results.[23] AdSense, on the other hand, includes

---

[18] Google provides financial information from 2008 Q1 – 2015 Q4, of which 2015 Q4 is a forecast. GOOG-00103813.

[19] Gold 12/11/2015 Dep., p. 69:15-22.

[20] Google provides financial information from 2008 Q1 – 2015 Q4, of which 2015 Q4 is a forecast. GOOG-00103813.

[21] Gold 12/11/2015 Dep., pp. 69:23-70:2; GOOG-00133931 at 940; "Find Free Content & Preview Paid Content," Google, https://support.google.com/googleplay/answer/2851613?hl=en.

[22] GOOG-00022386.

[23] Gold 12/11/2015 Dep., pp. 88:20-90:5; Agarwal 4/8/2011 Dep., pp. 76:23-77:25.

revenues generated from the search ads on mobile devices that appear on third-party websites based on the website's content and Google's algorithm for which ads would be of interest to users.[24] Display advertising revenues include revenues derived from display ads that appear on third-party websites.[25] Display includes ads that are matched to websites and mobile apps when keywords are related to a site's content or the interest of a user browsing a site, rather than the search terms used.[26] For Android to recognize Display revenues, users must navigate to mobile websites where display ads are shown; revenues are recorded on either a pay-per-impression or pay-per-click basis.[27]

30. In order to compile Android Ads revenues for earlier periods, historical P&L documents were used. For the 2008-2010 period, Android Ads revenues is composed of revenue from Distribution and Organic Ads, AFMA Ads, AFMC Ads, and AFMS Ads.[28] Organic Ads refer to ads generated from a user using an Android device and a Google search.[29] Distribution Ads, on the other hand, refer to ads served through a Google partner.[30] AFMA refers to AdSense for Mobile Applications and is part of the Display category; these ads appear within an app on a mobile device.[31] AFMC refers to AdSense for Mobile Content and is part of the Display category; these refer to ads that appear on websites that people are navigating to

[24] Gold 12/11/2015 Dep., p. 91:2-7; Lin 12/14/2015 Dep., pp. 139:2-25, 141:6-147:17.

[25] Gold 12/11/2015 Dep., p. 91:8-21.

[26] "About Display Network Only Campaigns," Google, https://support.google.com/adwords/answer/6340468; Lin 12/14/2015 Dep., pp. 40:16-41:20.

[27] Gold 12/11/2015 Dep., p. 93:10-16.

[28] GOOGLE-00395614.

[29] Agarwal 4/8/2011 Dep., pp. 76:23—77:25.

[30] Id.

[31] Agarwal 4/8/2011 Dep., p. 99:9-14; Interview of Jonathan Gold; Gold 12/11/2015 Dep., p. 31:13-24.

on mobile devices.[32] AFMS refers to AdSense for Mobile Search and is part of the AdSense category.[33]

31. Data on Android Ads revenues by category of advertising revenue are available for the relevant time period from 2008-2015 Q2, with the exception of 2011. In order to estimate Android Ads revenues for AdSense and Display in 2011, revenue is compiled from available total Android Ads revenue and Android Search revenues.[34] AdSense and Display revenues for Android are estimated based on the average ratio of Android AdSense to Display revenues in 2010 and 2012.[35]

E. Cost of Sales ("COS")

1. Traffic Acquisition Costs Paid to Google Advertising Partners

32. TAC stands for "traffic acquisition costs," and refers to the portion of ad revenue that Google pays to partners and websites.[36] Google makes these payments under agreements with the partners and websites that allow or facilitate Google to reach users with advertisements.[37] These payments are dependent on agreements between Google and third parties.[38]

---

[32] Interview of Jonathan Gold; Gold 12/11/2015 Dep., p. 31:13-24.

[33] Gold 12/11/2015 Dep., pp. 31:13-32:14; Interview of Jonathan Gold.

[34] GOOG-00022388; GOOG-00132245 at 247; GOOG-00132625.

[35] Exhibit 1c.

[36] "Financial Statements Glossary," Alphabet, https://abc.xyz/investor/other/additional-financial-information.html.

[37] "Traffic Acquisition Cost – TAC," Investopedia, http://www.investopedia.com/terms/t/traffic-acquisition-cost-tac.asp.

[38] Gold 12/11/2015 Dep., p. 24:10-22.

33. TAC data associated with ads served on Android devices are available from 2008 to 2010, but due to a change in financial reporting, Google no longer records TAC separately for advertising served on Android devices in the ordinary course of preparing its P&Ls.[39] Accordingly, in order to estimate TAC for Android from 2011 to 2015, I used the ratio of Google's TAC to Revenue for Adwords, AFS, and Display; this is in accordance with Google's methodology of calculating Android TAC amounts based on the total traffic acquisition costs for Adwords, AFS, and Display.[40] By comparison, Mr. Malackowski relies on the ratio from Google's Total Network Member TAC as a percentage of total Google Ad revenue, which he applies to Android advertising revenue, as well as information from the deposition of Jonathan Gold.[41] As a result of his calculation, Mr. Malackowski generates TAC rates between [redacted] relative to the TAC rates of between [redacted] that I calculate.[42] For comparison, historical TAC rates as a share of Android revenue for which there is information available are [redacted], [redacted], and [redacted] for 2008, 2009, and 2010 respectively.[43]

2. 2011-2015 COS – Hardware, Apps, Digital Content, and Infrastructure & Other COS

34. Due to a change in the Android business unit's internal evaluation of its financial data, the COS categories for Google's internally-maintained Android P&L changed between

---

[39] Interview of Jonathan Gold.

[40] Exhibit 1d; Interview of Jonathan Gold.

[41] Malackowski 1/8/2016 Report,¶ 299. However, Mr. Malackowski misapplies the [redacted] TAC rate stated in Jonathan Gold's deposition. The [redacted] TAC rate referred to Android search advertising rather than overall TAC rates for Android (which include Search, AdSense, and Display). See Gold 12/11/2015 Dep., p. 188:13-24; GOOG-00130338 at 343.

[42] Exhibit 1a, Exhibit 1d, and Malackowski 1/8/2016 Report, Exhibit 7.1.

[43] GOOGLE-00395614.

2010 and 2011. From 2011 to 2015, Google has recorded COS categories segmented by Hardware, Apps, Digital Content, and Infrastructure and Other COS.

35. Hardware COS is included for the period from 2011 through 2015.[44] Hardware COS includes the price paid for phones, tablets, watches and accessories, shipping costs, credit card fees, warranty costs, RMA, and E&O expenses.[45]

36. Apps COS is included for the period from 2011 through 2015.[46] Apps COS includes payments for developers, credit card fees and payments to carriers.[47]

37. Digital Content COS are included for the period from 2011 through 2015.[48] These costs include payments to carriers, credit card companies and content owners, such as publishers.[49]

38. Infrastructure and other COS are included from 2011 through 2015.[50] Infrastructure and other COS primarily includes network costs allocated to Android or Google Play. In addition, other COS includes customer support costs and amortization of goodwill related to Android.[51]

---

[44] Google provided financial information from 2011 Q1 - 2015 Q4, of which 2015 Q4 is a forecast. GOOG-00103813.

[45] Interview of Jonathan Gold.

[46] Google provided financial information from 2011 Q1 - 2015 Q4, of which 2015 Q4 is a forecast. GOOG-00103813.

[47] Gold 12/11/2015 Dep., pp. 71:4-74:25.

[48] Google provided financial information from 2011 Q1 - 2015 Q4, of which 2015 Q4 is a forecast. GOOG-00103813.

[49] Gold 12/11/2015 Dep., p. 72:2-11.

[50] Google provided financial information from 2011 Q1 - 2015 Q4, of which 2015 Q4 is a forecast. GOOG-00103813.

[51] Gold 12/11/2015 Dep., p. 77:9-25.

3. 2008-2010 COS – Operations and COS (Including DTC)

39. In prior Android financial reports through 2010, COS categories were included for Operations and other COS. Operations costs include data center, network, and SRE costs. Data center costs include the expenses associated with data center servers storing relevant information.[52] Network costs include bandwidth costs to enable interaction between servers or servers and users.[53] SRE refers to site reliability engineers that are responsible for the data center and networks.[54]

40. The Android P&L also includes a line item expense for other COS. These costs largely include customer support expenses and the cost of selling hardware such as Nexus.[55]

F. Operating Expenses ("OpEx")

1. Direct Android OpEx

a. Engineering PM – Android Development

41. Operating expenses for Engineering PM are included under direct Android operating expenses. These include the salaries of product managers and engineers as well as contractors for Android. Engineering PM expenses also include TVC and equipment-related expenses.[56]

---

[52] Gold 12/11/2015 Dep., p. 39:14-20.

[53] Gold 12/11/2015 Dep., p. 39:21-25.

[54] Gold 12/11/2015 Dep., p. 40:13-21.

[55] Gold 12/11/2015 Dep., p. 40:1-2.

[56] Interview of Jonathan Gold.

b. Marketing

42. The Marketing costs are comprised of salaries for marketing employees as well as direct marketing expenses.[57] Salaries of marketing personnel are allocated to Android based on annual and semi-annual employee surveys in addition to personnel costs for marketing teams that are fully dedicated to Android.[58] In addition, Marketing includes expenses related to outside agencies and consultants.[59] Marketing also includes co-marketing expenses which are purchase orders associated with marketing done in conjunction with a third party.[60]

c. Legal, Sales, and Other

43. Legal expenses include internal legal personnel costs, outside counsel expenses and costs related to legal settlements.[61] The Sales line item includes the employee costs of sales and business development people working on Android-related activities; these are allocated to Android based on annual and semi-annual surveys.[62] Additionally, Sales expenses include T&A and outside professional services requested by the Sales team.[63]

2. Incremental Google General and Administrative ("G&A") OpEx

44. Additionally, I have determined that certain Google G&A OpEx—Finance, POps, and REWS—are incremental with respect to Android and thus should be attributed to the Android P&L. As an accounting matter, Google does not allocate G&A expenses across lines of

[57] Gold 12/11/2015 Dep., p. 43:6-10.

[58] Interview of Jonathan Gold.

[59] Interview of Jonathan Gold.

[60] Gold 12/11/2015 Dep., p. 44:4-18.

[61] Gold 12/11/2015 Dep., p. 80:7-19. I conservatively deduct [redacted] for legal expenses related to the previous Google-Oracle litigation.

[62] Gold 12/11/2015 Dep., pp. 43:18-44:2; Interview of Jonathan Gold.

[63] Interview of Jonathan Gold.

business. However, a portion of the G&A expenses represent additional costs associated with the Android group. In order to estimate this portion, the ratio of Android engineer headcount to total Google engineer headcount is used.[64]

a. Finance

45. The Google G&A expenses for the Finance group include costs associated with finance and accounting.[65] As discussed above, the portion of these expenses that are incremental with respect to Android is based on the ratio of Android engineer headcount to total Google engineer headcount.[66]

b. POps

46. Google POps G&A expenses include costs associated with HR and training operations.[67] The portion of these expenses that are incremental with respect to Android is based on the same ratio of Android engineer headcount to total Google engineer headcount.

c. REWS

47. The REWS group includes costs associated with real estate and other costs associated with shuttles, meals, micro-kitchens, and other related expenses.[68] The portion of these expenses that are incremental with respect to Android is based on the same ratio of Android engineer headcount to total Google engineer headcount.

---

[64] Note that unallocated G&A expenses are only available from 2009-2015. Google engineering headcount is categorized under the R&D team at Google. Exhibit 1e.

[65] Interview of Jonathan Gold.

[66] Gold 12/11/2015 Dep., pp. 81:23-82:3.

[67] Interview of Jonathan Gold.

[68] Interview of Jonathan Gold.

3. Incremental Google Search and Advertising Product Area OpEx

48. Finally, I have also determined the portion of OpEx from the Google Search and Advertising Product Areas that is incremental with respect to the ads that are delivered on Android devices. I analyzed the historical relationship between operating expenses in the Search and Advertising Product Areas and ad revenues over the 2003 to 2015 time period. The ratio of Search and Advertising operating expenses to ad revenues has stayed approximately constant over the entire period as ad revenues have grown substantially.[69] That means that Search and Advertising operating expenses have increased roughly proportionally with ad revenue. This, in turn, implies that Search and Advertising operating expenses increased by approximately [redacted] for every $1 of revenue from ads delivered on Android devices.[70]

G. Opportunity Costs Associated With Foregone Advertising on Non-Android Devices

49. The "opportunity cost" of taking a particular action is the value of the opportunities that are foreclosed and thus given up in taking the action. Economics teaches that opportunity costs should be accounted for in analyzing the profit of taking a particular course of action.[71]

50. As discussed above, Google earns revenue and profit from ads placed on non-Android handsets and other devices. When Google made the Android operating system available to OEMs and device manufacturers, and those OEMs and device manufacturers then

[69] Exhibit 1g.

[70] I used a regression of Search and Advertising operating expenses on ad revenues (deflated to real dollars) and adjusted for the number of days per quarter to obtain the measure of incremental Search and Advertising operating expenses per dollar of revenue from ads delivered on Android devices that is used in my calculations. A more complex vector error correction approach yields a slightly higher figure.

[71] R. Brealey and S. Myers, Principles of Corporate Finance (5th Ed., 1996), pp. 115-116.

released Android-based smartphones and tablets, this provided consumers with an alternative to other devices that were on the market or would have been brought to market in the absence of those Android devices. Specifically, the availability of Android devices reduced the number of consumers using other devices—in particular the iPhone—and thereby reduced the revenue and profits Google earned from advertising served on those competing devices. This is an opportunity cost to Google of making Android available.[72]

51. Based on an analysis discussed below, I determined that Google would recapture at least [redacted] of its ad revenue on Android handsets with ad revenue on iPhones. Applying this recapture rate to Android ad revenue yields the incremental ad revenue that Google would have made on the iPhone in the absence of Android. Similarly, applying the recapture rate to Android's incremental search and advertising expenses and TAC yields the incremental expenses that Google would have incurred on these revenues. A Google internal analysis noted that Google earns a lower profit margin on search ads on iOS devices than on Android devices.[73] Therefore, I have adjusted the profit margin on the recaptured revenue to account for this fact. Subtracting the adjusted costs from the recaptured revenue yields the opportunity cost

[72] As Mr. Malackowski notes at ¶¶ 67-91 of his report, at the time Google released Android, there was a strong demand among consumers for smartphones and tablets as a result of broad market factors having nothing in particular to do with Android or Google, or any particular operating system or device. Mr. Malackowski points to general market factors such as the growth of wireless connectivity, and the growth of mobile data usage and mobile gaming, as contributing to the growth of smartphone platforms. Google would have benefited to some degree from this strong demand for smartphones, by providing services and serving advertisements on other existing smartphone platforms, even had Android never existed.

[73] GOOG-00130338 at 343.

associated with ads placed on iOS devices.[74] This opportunity cost is properly subtracted from the Android-related operating profit.

H. The Android-Related Profit After Deducting the Applicable Android-Related Costs from Android Revenues

52. In the previous sections I have described my methodology for constructing the Android P&L, which contains all of the Android-related revenues and expenses. Not deducting the iPhone opportunity cost, I calculate total Android-related profits of [redacted].[75] Deducting the iPhone opportunity cost, I calculate total Android-related profits of [redacted].[76] I also calculate the Android-related profits excluding ad revenue and costs entirely to be [redacted].[77]

53. In the next sections, I discuss my calculation of the proper apportionment of the Android-related profits to the use of the allegedly infringing material. Again, this analysis assumes, without conceding, that Mr. Malackowski has shown a causal nexus between the allegedly infringing material and the revenues and profits identified in his report.

VI. Apportionment

54. I understand that, from a legal perspective, in calculating unjust enrichment damages in a copyright infringement case, the profits of the alleged infringer must be apportioned to subtract out any aspects of those profits attributable to factors other than the alleged infringement.

---

[74] Exhibit 1b.

[75] Exhibit 1a.1.

[76] Exhibit 1a.3.

[77] Exhibit 1a.2.

55. Apportionment in this context therefore involves isolating the portion of the profits that should be attributed to the alleged infringement rather than to other factors that contributed to the profits, but arose from events and occurrences other than the alleged infringement. The portion of profits attributed to the alleged infringement should reflect the relative economic importance of its contribution as compared to the contributions of these other factors.

56. When performing an apportionment, one must first properly delineate the boundaries of the alleged infringement so as not to attribute to the alleged infringement contributions to profit that were actually made by other factors. In this case, I understand that Oracle claims that the alleged infringement relates to the SSO and declaring code of 37 specific API packages in Android. I understand that the alleged infringement is limited to this allegedly infringing material. The alleged infringement does not encompass Google's choice to use the Java programming language in the Android operating system, and Oracle admitted at the earlier trial of this matter that the Java language is free for anyone to use.[78] Neither does the alleged infringement encompass the overwhelming majority of the Android platform, which includes a Linux kernel, a runtime, an applications programming framework, and core libraries apart from the 37 API packages at issue—all of which were created by Google or used by Google under appropriate licenses and contributed substantially to the success of Android—or the contributions of others, such as OEMs that designed Android handsets that were attractive to users. Finally, although Oracle is asserting a claim to Google advertising revenue on Android devices, the infringement alleged by Oracle does not encompass any aspect of Google's search

[78] Trial Tr. 939:2-12 (Ellison).

or advertising software, which existed prior to the introduction of Android and function independently of and do not depend on Android. None of the advertising revenues earned by Google from Android devices could or would exist without the additional, non-accused functionality Google developed, over a period of many years, in its search and advertising software. Accordingly, an economically valid apportionment must isolate the contribution to profit of the alleged use of the allegedly infringing material from the contributions to profit of the use of the Java language, the many other, non-accused components of the Android platform, and all the other factors that contributed to Android's success, such as other functionalities of Android, Google's innovation and investments in Android, ad, and search technologies, device manufacturers' innovation and investments in device development, etc.

57. In the sections that follow, I first discuss the various other factors that contributed substantially to Android's success and the profits at issue. The existence of such factors suggests that the apportionment to the allegedly infringing material should be, at best, small. I next identify the nature of the contribution of the alleged infringement and demonstrate that the economic evidence supports the conclusion that this contribution was limited. Finally, I provide several conservative approaches to quantifying the portion of the Android-related profits that should be attributed to the alleged infringement.

A. The 37 API Packages at Issue

58. There are 51 "Java" API packages included in Android; Oracle did not develop and does not own the copyrights on 14 of them, and these 14 API packages are not asserted

against Google here.[79] That leaves 37 "Java" API packages that are at issue in this case. Google's technical expert Dr. Astrachan testified that the 37 API packages "are a very small part of the Android Platform, kind of a very small part of the core libraries that's a very small piece of the Android Platform."[80]

B. Contribution of Google's Advertising and Search Technologies and Services

59. Google offers one of the world's most well-known and most popular internet search services. A user with a question of interest submits search terms to Google's search engine. Google returns search results in the form of a list of web pages relevant to the user's search terms. Google first developed it search service in the late 1990s; since then, Google has continually improved the search algorithm and added innovative features.[81] Google search was already well-established in the market by the time Android was launched.

60. Google offers advertisers the ability to place ads on Google search results pages based on a user's search terms so that advertisers can better target their ads. Google also offers a service whereby advertisers can place ads on third party websites also potentially based on the websites' visitors' search terms.

61. Google had been generating revenue from its ad technologies and services since 2000.[82] Google's advertising business was already a success on personal computers and mobile

[79] Trial Tr. 2180:3-19 (Astrachan). I understand that Oracle initially accused 51 Android API packages of copyright infringement, but reduced that figure to 37 due to the fact that it did not own any copyrights on 14 of the original 51.

[80] Trial Tr. 2182:8-12 (Astrachan).

[81] "Our History in Depth," Google, https://www.google.com/about/company/history/.

[82] Id.

devices, including the iPhone, prior to the Android launch.[83] As stated during its Q3 2015 Earnings Call, on-going improvements to mobile search and advertising technology, such as "improvement in ad formats and delivery" have driven increased mobile search revenue.[84]

62. Google uses the same search and ad technologies to display advertisements to a user of Google search on Android as it does to a user of Google search on a PC.[85] Most advertisers run their ads on both PC and mobile devices, although there might be variations in the ad specific to whether it is served on a PC or mobile device.[86] Google serves ads on other mobile devices, such as the iPhone.[87]

63. The percentage of the Android-related ad profits that should be apportioned to Google's search and ad technologies and services can be determined based on the portion of ad revenue that Google retains when its ads are viewed using non-Android mobile devices. Consider each party's contribution to the revenue that is generated when an advertisement is viewed in Google search results on a non-Android handset. In that situation, Google's contribution is confined to its ad and search technologies and services. The contributions of the vendor(s) responsible for the handset and operating systems, by contrast, are collectively everything else: the hardware (design, manufacturing, supply chain, etc.), the operating system (including analogues to the APIs in the 37 packages at issue in this case as well as many other functionalities), other software (e.g., app store and other applications), the strength of the

---

[83] Wojcicki 8/31/2011 Dep., p. 154:2-17.

[84] Alphabet (GOOG) Q3 2015 Results – Earnings Call Transcript, October 22, 2015, Porat at 2, http://seekingalpha.com/article/3596706-alphabet-goog-q3-2015-results-earnings-call-transcript.

[85] Wojcicki 8/31/2011 Dep., p. 94:2-9.

[86] Id. at pp. 97:1-99:13.

[87] Id. at p. 120:20-121:4.

vendors' brands, their promotional and marketing efforts, etc. An example is the iPhone, where Apple is the vendor of both the hardware and operating system. A Google internal analysis conducted in the ordinary course of business concluded that Google earns an annual search ad margin of [redacted] per Android phone, which given the 15% TAC on search ads on Android, implies an annual search ad revenue of [redacted] ([redacted]).[88] The document notes that the [redacted] search ad margin would be decreased to [redacted] on an iPhone. Thus, when Google places a search ad on an iPhone, [redacted] is what Google receives for its contributions, i.e., search and ad technologies and services. Thus, [redacted] ([redacted]) of search ad revenue on Android should be attributed to Google's search and ad technologies and services. Conversely, the remaining [redacted] of search ad revenue should be apportioned to Android. The same Google internal analysis notes that the margin on non-search ads is the same on Android as on iPhone. Thus, none of the non-search ad revenue should be apportioned to Android.

64. I calculate the apportionment of Android-related profits to Android as follows. I calculate apportioned revenue as [redacted] of search ad revenue plus [redacted] of hardware, digital content, and app store revenue. I calculate apportioned expenses as [redacted]% of hardware, digital content, and app store COS plus [redacted] of Android operating expenses (this excludes TAC and incremental ad and search OpEx, which are apportioned to the search and ad technologies and services). Taking the difference between the apportioned revenues and apportioned costs, I obtain [redacted] as the apportionment of the Android-related profit to Android (with the remainder being apportioned to the search and ad technologies and services).[89]

[88] GOOG-00130338 at 343.

[89] Exhibit 1a.4.

65. In a second step, the Android profit figure of [redacted] is itself apportioned to all of the various functionalities of Android and Android handsets, of which the allegedly infringing material constitutes only a small part, as described below.

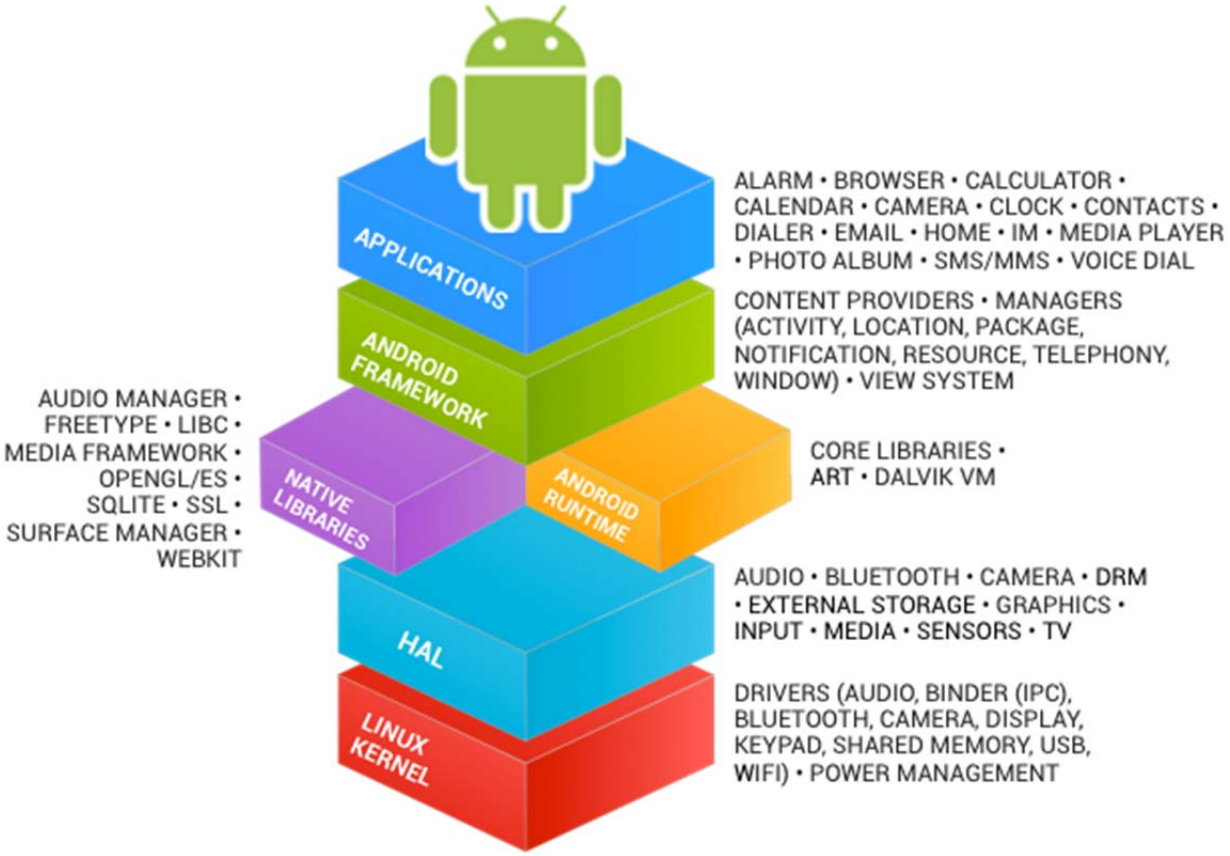
C. Reasons for the Success of Android Other Than the Alleged Infringement

1. Google's Efforts

66. Google has been the main driver behind the initial launch and continued development of Android. Android launched as version 1.0 and is currently on version 6.0.[90]

67. As pictured below, Android consists of a Linux kernel, a hardware abstraction layer, native libraries, Android runtime, the Android framework, and a set of built-in applications.

[90] For my understanding of the technical aspects of Android described in this section, I rely on the expert reports and testimony of Google's technical experts in this phase and in the previous trial. I understand that the code related to the 37 API packages at issue underwent little change after Android's launch. Kemerer 1/8/2016 Report ¶ 110. Therefore, all or nearly all of the value Google added to Android in versions subsequent to 1.0 involved functionality other than the 37 API packages.



Source: https://source.android.com/source/index.html

a. Linux Kernel

68. The Linux kernel is the portion of the Android operating system that interacts with and runs the hardware—the physical phone or tablet used by the consumer, including the camera, speakers, and display screen. Linux is available on an open source basis. However, "Android uses a version of the Linux kernel with a few special additions such as wake locks (a memory management system that is more aggressive in preserving memory), the Binder IPC driver, and other features important for a mobile embedded platform."[91] In other words,

[91] "Android Interfaces and Architecture," Android, https://source.android.com/devices/index.html.

Google has made contributions to the open-source Linux kernel that provide additional, customized functionality for Android.[92]

b. Hardware Abstraction Layer

69. The hardware abstraction layer (HAL) provides an interface between the software APIs and the hardware. Specifically, it "defines a standard interface for hardware vendors to implement and allows Android to be agnostic about lower-level driver implementations. The HAL allows you to implement functionality without affecting or modifying the higher level system."[93]

c. Native Libraries, Core Libraries, and the Android Framework

70. The native libraries, core libraries, and Android Framework provide APIs that can be accessed by applications that run on Android.[94] The native libraries are written in the C++ programming language, not in Java. The core libraries are "Java" based, although some of the "Java" APIs are just "wrappers" around a C++ API. In other words, when one of these APIs is called, the computational work is done by C++, not Java. The Android Framework provides an easy-to-work-with environment for developers.[95]

71. The 37 API packages at issue in this case are part of the core libraries. However, Google provided over 131 other API packages to the Android native libraries, core libraries, and Android Framework. Google has continued to innovate, continually adding large numbers of

[92] "Android A to Z: What is a Kernel? Android Central, January 23, 2012, http://www.androidcentral.com/android-z-what-kernel.

[93] "Android Interfaces and Architecture," Android, https://source.android.com/devices/index.html.

[94] Trial Tr. 2161:10-2171:22 (Astrachan).

[95] Interview of Tim Bray; "NARRATIVE: Mobile + Android, last updated: 03/18/08," GOOGLE-23-00000001-027 at 003. Also see "Android 101: An Introduction to Android and Android Partnerships, Last Updated: December 2008," GOOGLE-00298438-484 at 466.

features to Android and corresponding APIs. For example, with Lollipop, Google added approximately 3,000 new APIs.[96]

d. Android Run Time

72. Applications are computer programs that are compiled and then run on an Android device. Initially, Google designed Android so that each application written in the Java programming language runs in what was called the "Dalvik" virtual machine. I understand that the Dalvik virtual machine is not subject to any of the copyrights asserted in this case. Rather, aspects of Dalvik were accused by Oracle in the first trial of this case of infringing certain Oracle patents, but Google prevailed as to those issues, obtaining a verdict of no infringement from the jury. Dalvik was designed to be efficient and compact,[97] which can be important given the mobile device context.

73. Google subsequently developed "Android Run Time" (ART) to replace Dalvik. Under ART, applications are compiled to machine code upon installation. This provides benefits, including improved performance of the application.[98]

e. Applications Layer

74. Android provides a large number of functionalities through built-in applications. These include the software that runs a web browser, email, calendar, phone, text messaging, camera, calculator, and flashlight, among others. These built-in applications provide users with a substantial amount of functionality even in the absence of any third party apps.

---

[96] Lockheimer 12/8/2015 Dep., p. 271:5-15.

[97] Ghuloum 12/9/2015 Dep., pp. 155:2-156:24.

[98] "ART and Dalvik," Android, https://source.android.com/devices/tech/dalvik/.

f. Support for Third Party Application Developers

75. Google provides applications developers with tools, such as the Android Suite, that assist in the app development process. Google also provides plug-ins so that developers can use other popular application development tools, such as Eclipse, to develop Android applications. Google also has a team of "developer advocates" that provide support and assistance to third party app developers.[99]

g. Support for Applications Programming in C++, C, and JavaScript

76. Google has provided ways for developers to write applications in programming languages other than Java, and many developers, who prefer using languages other than Java for various reasons, have taken Google up on this offer. For example, the Native Development Kit (NDK) allows a developer to write an Android application in C or C++.[100] A substantial number of applications, including many of the most popular applications offered through the Google Play store, were in fact developed using the NDK.[101] I assembled a list of apps that appeared in the daily top 100 download lists for both Android and iOS during the period 2012 to 2013. Google personnel identified whether each app was written in Java or C/C++. Of the 188 apps on the list for which the language can be determined, 162 were written in C/C++.[102] Mr. Ghuloum testified that Google has "done a lot of work to really bolster the quality of our C and C++ [framework] in the last few years."[103]

---

[99] See generally Rutledge 12/9/2015 Dep.; interview with Tim Bray.

[100] Ghuloum 12/9/2015 Dep., p. 63.:8-17

[101] Id.

[102] Exhibit 2b.

[103] Ghuloum 12/9/2015 Dep., p. 68:2-9.

77. Google has also developed RenderScript, "a framework for running computationally intensive tasks at high performance on Android" where code is written in a variant of the C programming language.[104] RenderScript "is especially useful for applications performing image processing, computational photography, or computer vision."[105]

78. Google also provides support for the use of JavaScript in Android in certain contexts.[106]

h. Google Mobile Services and Google Play Services

79. Google contributed many popular Android applications that were important for getting the Android ecosystem off the ground.[107] Today, "Google Mobile Services" is a package of apps that are provided pre-loaded on many Android devices, including Google Maps, Gmail, YouTube, and Google Play.[108]

80. Google also has developed a set of functionalities called "Google Play Services." These are a set of APIs that are not part of Android, but instead sit on top of Android and provide app developers with tools related to the development and optimization of app functionality, such as the ability to insert a Google map within an app.[109]

---

[104] Ghuloum 12/9/2015 Dep., p 63:8-17; Android, http://developer.android.com/guide/topics/renderscript/compute.html.

[105] Id.

[106] Ghuloum 12/9/2015 Dep., p. 63:18-23.

[107] "Google Android, OC Quarterly Review – Q4 2010," October 12, 2010, GOOGLE-01-00053552-591 at 566.

[108] Lin 12/14/2015 Dep., pp. 73:8-76:2. While these apps may have been written in the Java programming language, as discussed below, Google had the resources, capabilities, and incentives to write these applications in another language. Consistent with this, Google has written most of these apps in Objective C for use on the iPhone.

[109] Lockheimer 12/8/2015 Dep., pp. 253:1-257:5. The Google Play Services APIs are not in any way related to the 37 APIs at issue in this case. Google does not earn revenue from Google Play Services, which are designed to

i. Google's Hardware Development, Promotional Efforts, and Brand Name

81. Google has designed and sold a series of mobile devices that run the Android operating system. To date, Google has sustained a net loss on these products.[110] However, Google's primary purpose in developing these devices was not to earn a profit, but instead to use them as a showcase for Android's capabilities.[111] Thus, Google's efforts on hardware development are an investment in the success of the Android platform.

82. Google has invested in the advertising and promotion of Android to build awareness among users.[112]

83. The fact that Google is behind Android has been a powerful driver of Android's success. The Google brand name built confidence in Android among app developers, OEMs, and users.[113]

j. Support for Device OEMs

84. Google has provided substantial support for OEMs who develop Android devices.[114] As discussed below, the wide variety of attractive Android devices has been an important cause of Android's success.

---

make it easier and more attractive for third party developers to develop Android apps. Gold 12/11/2015 Dep., pp. 251:9-252:4, 255:4-17.

[110] Exhibit 1a.

[111] "Nexus Phones Will Never See Huge Sales – But Here's Why They Don't Need To," Fortune, September 30, 2015, http://fortune.com/2015/09/30/google-nexus-smartphones-about-innovation-not-sales/.

[112] "Android OC Quarterly Review – Q4 2010," October 12, 2010, GOOGLE-01-00064207 at 228; "Android OC Quarterly Review – Q3/Q4," GOOGLE-01-00072883 at 894; "Introduction to Android," May 2015 GOOG-00130338 at 360.

[113] "Why is Google Android Beating Symbian?" CNET, November 16, 2009, http://www.cnet.com/news/why-is-google-android-beating-symbian/ "Google Android Strategy."

[114] Lockheimer 12/8/2015 Dep., pp. 158:19-159:19.

k. Google Play App Store

85. Google has provided and managed an app store that serves to connect app developers with users. The app store provides information to users, such as top download lists, user reviews, etc. that assist users in deciding which apps to download. The app store provides app developers with a relatively low cost way to reach users and thus reduces the cost of the distribution of an app.[115]

2. Google's Decisions to Make Android Free and Open Source

86. Android is an open source and free platform, which has encouraged adoption by handset manufacturers and applications developers and led to the emergence of an "ecosystem."[116] As Google has stated, "[w]e credit Android's rapid adoption to the fact that we made it available under an open source license."[117] Making Android open source facilitated Google entering into partnerships with multiple handset manufacturers and carriers.[118]

87. As discussed below, the open source nature of Android allowed handset OEMs to add functionalities to differentiate their products. An OEM could even use Android as the

[115] "Publishing Your First App in the Play Store: What You Need to Know," Android Authority, May 20, 2014, http://www.androidauthority.com/publishing-first-app-play-store-need-know-383572/.

[116] "Android Tipped to Overtake iPhone by 2012," VNUNet United Kingdom, March 6, 2009; "Finding Value in Android," Bank of America Merrill Lynch, September 9, 2010, GOOGLE-01-00048436-484 at 438, 444; "Google's Android Mobiles Overtake Global iPhone Sales," Financial Times, August 12, 2010, http://www.ft.com/cms/s/2/77ed3ddc-a63f-11df-8767-00144feabdc0.html#axzz41IpKBZeW; "Google's Android Ambition is to Reshape the Mobile Industry, Report Says; But Android Faces Big Problems, Tight Deadlines, Says Report," Network World Fusion, January 3, 2008 (TX 3110); "Eric's question # 14 - Mobile Strategy 2007," GOOGLE-01-00024675-716 at 676; OAGOOGLE0018885324.

[117] "Google Android, OC Quarterly Review – Q4 2010," October 12, 2010, GOOGLE-01-00053552-591 at 562.

[118] "Android on Steroid: Google Enters Mobile Market with a Splash; Main Buy," Jefferies & Company, Inc., September 24, 2008, p. 4.

starting point for an operating system for a non-Android device, as Amazon did with the Kindle Fire.[119]

3. The Efforts of OEMs

a. Incentives and Economic Position of OEMs When Android Was Introduced

88. At the time of Android's introduction, Apple's iPhone was already a market success. Android offered handset manufacturers (OEMs) such as Motorola, Samsung, LG, and HTC a way to better compete quickly with the iPhone, and was preferable compared to alternative operating systems.[120]

89. Android was attractive to handset OEMs because, on the one hand, Android provided a full mobile stack with a core set of handset functionalities for free, but, on the other hand, the open source nature of Android allowed an OEM to develop additional software with additional functionalities with which it could differentiate itself from its handset competitors.[121] For example, Samsung developed its own user interface to run on top of Android, which gives its devices a different look than devices from other manufacturers.[122] Similarly, Samsung developed software that provides users of the Note product the ability to use a stylus and

---

[119] Lockheimer 12/8/2015 Dep., p. 163:1-165:20.

[120] "Android: On a Bender, Telecom Equipment," Arete Research Services LLP, July 14, 2010, GOOGLE-01-00049780-784 at 780; OAGOOGLE0018885324. OAGOOGLE0018885324 ("Service providers are demanding Android based handsets (especially the ones that cannot get iPhone"; service providers are the handset manufacturers' customers); OAGOOGLE002778854 at 855 ("The Mobile industry is turning toward Linux-based platforms to combat Nokia dominance and prevent formation of another Microsoft"); OAGOOGLE002778877 (negatives of Windows Mobile).

[121] "NARRATIVE: Mobile + Android, last updated: 03/18/08," GOOGLE-23-00000001-027 at 003. Also see "Android 101: An Introduction to Android and Android Partnerships, Last Updated: December 2008," GOOGLE-00298438-484 at 466; Lin 12/14/2015 Dep., pp. 84:3-85:7.

[122] Lockheimer 12/8/2015 Dep., p. 294:14-295:3.

Samsung provided a fingerprint scanner before that capability was provided by Android.[123] Allowing OEMs to differentiate was a fundamental aspect of Google's strategy with Android.[124]

90. All else equal, carriers want to offer a wide variety of handsets to appeal to a broad range of customers.[125] In contrast to the iPhone, which had a limited number of models, as discussed above Android provided for a variety of handsets by encouraging many OEMs to launch their own Android handsets. Thus, carriers also had strong incentives to support Android.

91. The strong interest in Android among OEMs (and others) was demonstrated by the broad membership of the Open Handset Alliance. These members include: Google, Dell, eBay, HTC, Kyocera, Motorola, Samsung, Sharp, Accenture, Noser Engineering, Sony Ericsson, Aplix Corporation, Acer, Sprint, among others.[126]

b. OEMs' Hardware and Software Development Efforts

92. An important contribution to Android's success as an operating system has been the wide range of devices running Android that have been developed and introduced by OEMs such as HTC, Samsung, LG, and Motorola.[127] Some OEMs have invested in customizing Android or in adding software or hardware features that provide additional functionality on top of Android.[128]

---

[123] Id. at p. 296:2-15; Lin 12/14/2015 Dep., pp. 164:12-167:23.

[124] See, e.g., Lockheimer 12/8/2015 Dep., p. 173:9-21.

[125] Meier 12/11/2015 Dep., p. 95:2-9.

[126] "Members," Open Handset Alliance, http://www.openhandsetalliance.com/oha_members.html.

[127] Meier 12/11/2015 Dep., p. 39:20-40:6.

[128] Felix Lin 12/14/2015 Dep., pp. 69:13-70:4, 78:2-23, 164:12-166:6; Lockheimer 12/8/2015 Dep., p. 294:14-295:3.

93. As a result of OEMs' product development efforts and the competition between OEMs promoted by Android, the prices of Android devices have decreased substantially over time, while their quality has improved dramatically. For example, between 2010 and 2015, the quality-adjusted contract price for an Android handset decreased from $213 to $14 (the quality-unadjusted weighted average contract price decreased from $213 to $88), while thickness of the handset decreased by 38%, the handset's display size increased by 50%, and the megapixels of the handset's camera increased by 138%.[129] Handset variety has increased as well, with the number of available Android devices increasing from 15 to 102 over the period (although all devices may not be available in all geographies) and the number of manufacturers making Android handsets increasing from 3 to as many as 14.[130] The price decreases and quality improvements in available Android devices has made Android more attractive to users and contributed to Android's success.[131] Indeed, despite iOS having a lead over Android in terms of number of apps in the first several years after the Android launch, Android ultimately surpassed iOS in terms of market share. This is largely due to the attractiveness of the Android devices (as well as the functionalities of the Android operating system itself).

c. OEMs' and Carriers' Brand Names and Promotional Efforts

94. Along with promotional efforts by Google related to Android generally, OEMs also invested in promotion of their specific Android devices and the Android brand.[132]

[129] Exhibit 2a.

[130] Id.

[131] See, e.g., M.J. Kim, Essays on the Economics of the Smartphone and Application Industry, University of Minnesota (2013), p. 45.

[132] Meier 12/11/2015 Dep., p. 79:1-6.

4. Conclusion

95. It is notable that Google succeeded with Android as an operating system for mobile devices, whereas many other companies did not. In fact, several of the failed attempts involved operating systems where Java was the applications programming language. For example, in the early 2000s a company called SavaJe offered an OS for mobile devices with a Java SE programming language layer.[133] Despite having Java as the applications programming language, SavaJe did not have much success signing up OEMs.[134] In April 2007, almost 1.5 years prior to the launch of Android, Oracle's predecessor, Sun, acquired the intellectual property assets of SavaJe.[135] But, Sun abandoned the effort to offer a SavaJe-based mobile OS.[136] Sun and Oracle later initiated other efforts to build a Java-based OS for mobile devices, but again these efforts were abandoned.[137] This demonstrates that the contributions that Google brought to the table were substantially more important than the particular choice to use Java as an applications programming language (let alone the use of the allegedly infringing material).

D. Evaluating the Contribution of the Alleged Infringement

1. The Nature of the Contribution of the Alleged Infringement

96. Throughout this case, Oracle has contended that the SSO and declaring code of the 37 API packages (out of the 168 API packages in Android that are available to applications

[133] "Leading Java Software Vendors Endorse the SavaJe XE Operating System," PR Newswire, June 4, 2001, http://www.prnewswire.com/news-releases/leading-java-software-vendors-endorse-the-savaje-xe-operating-system-72019272.html.

[134] "Sun Buys Up SavaJe, But Motive Remains Unclear," RCR Wireless News, April 16, 2007, http://www.rcrwireless.com/20070413/archived-articles/sun-buys-up-savaje-but-motive-remains-unclear-2.

[135] Id.

[136] Trial Tr. 325:15-327:13 (Ellison).

[137] Trial Tr. 1913:6-1915:20 (Gering).

programmers) were valuable to Google, and contributed to the success of and profits resulting from Android, because those 37 API packages were "familiar" to Java programmers. Oracle further reasons that this familiarity led to an increased number of developers working on Android applications at any point in time and thus a greater number of Android applications at any point in time; a greater number of Android applications in turn potentially could have led to more Android users who would have generated more revenue and profits for Google via ads, app store revenue sharing, or hardware sales.

97. However, in actuality, as discussed below, the economic evidence demonstrates that these potential effects of "familiarity" with the 37 API packages were small at best, which would mean that the contribution of the alleged infringement to Android-related profits was small as well.

98. In addition to the theory that Google was able to attract developers to Android because of their supposed familiarity with the 37 API packages at issue, Oracle's expert Dr. Kemerer also asserts that the use of the allegedly infringing materials led to a greater degree of "stability" in Android because the 37 API packages at issue were well-established and thus, for example, required less debugging, and greater "stability" would make Android more attractive to developers and then, in turn, to users. This assertion is incorrect because it is based on faulty factual claims as well as incoherent logic:

- The 37 API packages at issue constitute only a small portion of the overall Android code base, and the SSO and declaring code of the 37 API packages at issue an even smaller portion. Even if the 37 API packages were "stable" in and of themselves, their use would in no sense guarantee stability of Android as a whole, which would be the relevant inquiry both for potential Android developers and consumers.

- In claiming that Java was "stable" as of 2006[138], Dr. Kemerer confuses "stability" with stagnation. As discussed elsewhere in this report, Sun's Java platform (and its APIs) did not change not because of "stability," but because Sun stopped innovating. Stagnation (lack of innovation) is not a positive and, in fact, this stagnation explains why Java's popularity was declining prior to the introduction of Android.
- By the same token, Dr. Kemerer's claim that Android is not "stable" because Google has added additional APIs to Android[139] confuses "instability" with innovation. Over time Google has added APIs to Android to provide additional valuable functionality that benefits OEMs and device manufacturers, wireless carriers, and consumers. This is innovation and innovation is a good thing (unlike Java's stability/stagnation). Importantly, however, Google generally has not changed existing APIs. Thus, existing applications continue to work on new Android versions.[140]
- Because he confuses "instability" with innovation, Dr. Kemerer inconsistently claims on the one hand that Java was not stable until 2006 while on the other pointing out that Java exhibited substantial user growth in the late 1990s.[141] If "instability" deterred users from using the platform, Java should not have enjoyed substantial user growth during the period it exhibited "instability" (according to Dr. Kemerer).
- Any "stability" that Java might have is not unique to Java. Many other programming languages offer similar "stability." As just one example, C++ is a well-established programming language with analogues to the 37 API packages at issue.
- In general, Dr. Kemerer seems to be implicitly comparing the stability of the 37 API packages to the stability of a hypothetical set of API packages that Google could have developed to be used in place of the 37 API packages. Even if such hypothetical API packages would have been less stable than the 37 API packages, Dr. Kemerer is just setting up an irrelevant strawman. Google's best alternative to using the allegedly infringing material was to use the OpenJDK or another programming language, not develop a set of APIs to replace the 37 APIs at issue.

---

[138] Kemerer 1/8/2016 Report ¶ 110.

[139] Kemerer 1/8/2016 Report ¶¶ 96-114.

[140] "Development Considerations," Android, http://developer.android.com/guide/topics/manifest/uses-sdk-element.html#considerations ("Android applications are generally forward-compatible with new versions of the Android platform. Because almost all changes to the framework API are additive, an Android application developed using any given version of the API (as specified by its API Level) is forward-compatible with later versions of the Android platform and higher API levels. The application should be able to run on all later versions of the Android platform, except in isolated cases where the application uses a part of the API that is later removed for some reason.").

[141] Kemerer 1/8/2016 Report ¶¶ 75, 110.

- In numerous instances, Dr. Kemerer confuses the implementation of an API with its SSO and declaring code.[142] For example, the need to debug—cited by Dr. Kemerer as a cause of "instability"—is unlikely to be an issue for SSO and declaring code. This is important because Google is not accused of copying implementations in this case (with one negligible exception).

2. Economic Evidence Demonstrates That the Profit Contribution of the Alleged Infringement Was Small

99. There are a number of economic reasons why the contribution of the alleged infringement to the Android-related profits was small.

a. Java Offered No Significant Advantage Over Other Existing Applications Programming Languages

100. Even assuming that the use of Java as an applications programming language was coextensive with use of the allegedly infringing materials, Google could have chosen to use (and, with respect to other aspects of the Android platform, in fact did choose to use) one of many other existing applications programming languages such as JavaScript, Python, or C/C++. Any advantage Java offered over these other languages was small and also accompanied by disadvantages.

101. Oracle's experts claim that Java was a popular language among programmers, but a number of other languages had similar popularity among programmers and were considered to be in the same "tier."[143]

---

[142] Kemerer 1/8/2016 Report ¶¶ 12, 95.

[143] A 2010 analysis found that C, Python, and JavaScript ranked higher than Java on one of two popularity metrics. "Ranking the popularity of programming languages," dataists, December 9, 2010, http://www.dataists.com/2010/12/ranking-the-popularity-of-programming-langauges/. A related 2011 analysis defined a "Tier 1" of languages in terms of popularity that included C, C++, C#, Java, JavaScript, Objective-C, Perl, PHP, Python, Ruby, and Shell scripts. "Revisiting the Dataists Programming Language Rankings," RedMonk, September 6, 2011, https://redmonk.com/sogrady/category/programming-languages/. According to the TIOBE index cited by Oracle's experts, C was approximately as popular as Java around the time of Android's launch. "TIOBE Index for February 2016," TIOBE,

102. While Oracle asserts that there are approximately 6 million Java programming language developers, it is not the case that all of these developers have developed applications for Android. Indeed, as of early 2011, there were only about 20,000 Android developers, and many of these write in C++ and use the NDK in developing applications for Android, rather than writing in the Java language.[144] By way of comparison, there were approximately 75,000 iPhone developers at that point in time, despite the fact that the iPhone used the less familiar Objective C language and not the Java language.[145] The large majority of Java programmers are writing for server and other enterprise applications as opposed to mobile applications.

103. An operating system supporting applications programming languages other than Java was and is hardly unprecedented. For example, at the time of Android's development, Symbian supported C++ (both a "Symbian" version of C++ and "standard" C/C++), Python, and .NET, among others (including Java ME); Microsoft Windows Mobile 2003 supported C++ (Embedded Visual C++); Palm OS supported C and C++; and, of course, Apple's iOS supported Objective C.[146] Mobile operating systems continue to support languages other than Java. For

---

http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html. A 2015 survey of developers (not restricted to mobile app developers) found that 68% of respondents were moderately or highly conversant with Java and 53% were moderately or highly conversant with C/C++. Developer Insights Report, IDC, August 2015, p. 26. This study notes that the Java number has benefited from Android, so the two numbers would have been closer at the time of Android's launch. Id. Only 36% were moderately or highly conversant in Objective C, but that has not hindered iOS app development.

[144] "App Genome Report – February 2011," Lookout Mobile Security. By early 2012, there were 68,000 Android developers. See "How many developers/companies have submitted an app to the iOS App Store or Android Market," Quora, February 5, 2012, https://www.quora.com/How-many-developers-companies-have-submitted-an-app-to-the-iOS-App-Store-or-Android-Market/answer/Philipp-Berner?srid=oWse&share=1.

[145] "App Genome Report – February 2011," Lookout Mobile Security.

[146] "Developing on Symbian, the World's Most Popular Mobile OS," DevX.com, July 22, 2010, http://www.devx.com/wireless/Article/45162; "Getting Started With Windows Mobile Application Development," Dr. Dobb's – The World of Software Development, September 22, 2006, http://www.drdobbs.com/mobile/getting-started-with-windows-mobile-appl/193004929; "Software

example, Microsoft Windows Phone supports C#, Visual Basic, C++, and JavaScript; Samsung's Bada supports C++; Firefox OS supports HTML5, CSS3, and JavaScript; Ubuntu Touch supports HTML5 and Qt/QML; and Apple's iOS supports Swift in addition to Objective C.[147]

104. As of the time of the introduction of Android, Java was viewed as being on the decline, in significant part due to the stagnation created by Sun's failure to develop and introduce new versions of its various Java platforms. A 2005 report based on a survey of developers concluded that "Java use is slipping" and that the percentage of North America programmers that used Java as one of their principal languages had been declining in recent years.[148] A 2010 report based on a survey of programmers concluded that "Java has a long-term downward trend" and attributed the downward trend in part to the fact that Oracle/Sun had not made improvements in Java.[149] Other industry observers similarly noted that Java was losing ground to other languages, in part due to Oracle/Sun's failure to make improvements,

---

Development for the Palm OS," Netmeister.org, 2000, https://www.netmeister.org/palm/PalmMisc/PalmMisc.html "The Swift Effect: Apple's New Programming Language Means Way More iPhone Developers and Apps," The Verge, June 2, 2014, http://www.theverge.com/apple/2014/6/2/5773928/apple-swift-programming-developers-objective-c. Nokia encouraged developers to write in C++, even though applications written in the Java programming language could be used on Nokia handsets. See "The Future for Mobile Java," Eriks Diary, May 9, 2010, http://eriksdiary.blogspot.com (May 9, 2010).

147 "Create Your First App," Microsoft, https://msdn.microsoft.com/enus/library/windows/apps/bg124288.aspx; "Samsung Releases Bada SDK in Adavance of First Bada Phone," BetaNews, May 7, 2010, http://betanews.com/2010/05/07/samsung-releases-bada-sdk-in-advance-of-first-bada-phone/; "9 Things You Should Know About Firefox OS," Hongkiat Technology Design Inspiration, May 3, 2014, http://www.hongkiat.com/blog/9-things-about-firefox-os/; "For Developers By Design," Ubuntu, http://www.ubuntu.com/phone/developers.

148 "Java? It's So Nineties," Bloomberg Business, December 12, 2005, http://www.bloomberg.com/bw/stories/2005-12-12/java-its-so-nineties.

149 "C is Number One!" ComputerWorld, April 7, 2010, http://www.computerworld.com/article/2468762/open-source-tools/c-is-number-one-.html; "Google Exec Worries Over 'Rudderless' Java," PC World, April 14, 2010, http://www.pcworld.com/article/194165/article.html.

and that this would ultimately result in less use of Java as a teaching language.[150] Personnel from Sun itself internally stated at around the time of Android's launch that "Java is considered legacy" and "fragmented."[151] The extent of Java fragmentation was noted in 2004 in a JavaWorld article: "you end up with numerous JVM versions in the marketplace, each running with different APIs on different operating systems. That is perhaps the least publicized, but most harmful of fragmentation elements…All of these differences add to the growing list of device idiosyncrasies that must be considered when programming a J2ME application."[152]

105. While Java had grown stagnant prior to Android, the launch of Android has been responsible for the recent resurgence of Java as a programming language. Mr. Schwartz, Sun's CEO, stated at the time of Android's launch that "Google and the Open Handset Alliance just strapped another set of rockets to the [Java] community's momentum."[153] This prediction was borne out. A 2015 report concluded that "after a year-and-a-half in second place, behind the C

[150] "The Future of Java," Forrester, January 23, 2011.

[151] Trial Tr. 1918:10-1919:4(Gering); Trial Tr. 1933:20-1935:3 (Rizvi).

[152] "The Fragmentation Effect," JavaWorld, May 24, 2004, http://www.javaworld.com/article/2072740/mobile-java/the-fragmentation-effect.html; "Inside Google's Android and Apple's iPhone OS as Core Platforms," Apple Insider, November 5, 2009, http://appleinsider.com/articles/09/11/05/inside_googles_android_and_apples_iphone_os_as_core_platforms ("The 'run everywhere' premise of Java ME is also complicated by the fact that different phones (even from the same vendor) implement the Java virtual machine differently. This results in user confusion as each app has to be tested and optimized for each new phone model…That's why Sun's Java ME platform, despite being touted as 'the most ubiquitous application platform for mobile devices across the globe,' hasn't resulted in a popular, successful market for smartphone software."); "Write Once, Run Anywhere Not Working for Phones," CNET, July 15, 2005, http://www.cnet.com/news/write-once-run-anywhere-not-working-for-phones/ ("…writing a [Java] program that can run on any handset still isn't possible…Sun's efforts to end the fragmentation problems have so far come up short."); "Java WORA Defences Spent," ComputerWorld UK, May 30, 2012, http://www.computerworlduk.com/blogs/simon-says/java-wora-defences-spent-3569627/ ("Most developers I've spoken with have no clue why Oracle thinks WORA [write once, run anywhere] is threatened [by Android]…[Java ME] has been fragmented for years, as each handset vendor has added their own unique approach to their own unique features. The fragmentation had nothing to do with Android and existed before it.").

[153] "Congratulations Google, Red Hat and the Java Community!" Jonathan's Blog, November 5, 2007, http://web.archive.org/web/20101023072550/http://blogs.sun.com/jonathan/entry/congratulations_google.

language, Java surged back into first place [in a "language popularity index"]." The report attributes Java's turnaround to Android: "Android's ongoing success is probably the major reason for Java's revival"; "[Java] has been in slow decline for many years now…[but] the increasing demand on Java Android programming has exceeded this decline."[154] Other industry observers similarly concluded that Java's role in Android application development boosted Java's standing and extended its life.[155] As a former Sun product manager for Java stated, "Java isn't exactly ubiquitous, but it's a lot closer – thanks in no small part to the advent of the Android OS."[156] Java was even recently named "programming language of 2015" by a prominent software consulting firm.[157] The consulting firm noted that Java is used in Android.

106. Many programmers are "familiar" with multiple languages. Thus, many Java programmers already are familiar with the other languages that Google could have chosen to use. For instance, according to the 2015 VisionMobile State of the Developer Nation report, programmers typically write in multiple programming languages.[158] An earlier 2010 report from VisionMobile similarly found that most developers work on multiple platforms (an average of 2.8 platforms per developer) and that "[a]mong developers taking the survey as iPhone

[154] "Java Regains Spot as Most Popular Language in Developer Index," InfoWorld Tech Watch, April 14, 2015.

[155] "Once Declining Java Cements Its Lead in Language Popularity Index," InfoWorld Tech Watch, August 7, 2015, http://www.infoworld.com/article/2909894/application-development/java-back-at-1-in-language-popularity-assessment.html; "Java's Key to Success is Simplicity," JavaWorld, May 20, 2015, http://www.javaworld.com/article/2925060/java-platform/javas-key-to-success-is-simplicity.html.

[156] "Java at 20 Years, Part 1: What's in a Name?" Application Development Trends Magazine, May 22, 2015, https://adtmag.com/blogs/watersworks/2015/05/java-at-20.aspx.

[157] "TIOBE Index for February 2016," TIOBE Software, February 2016, http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html.

[158] The report describes a survey of developers that asked about developers' primary and secondary languages. "State of the Developer Nation Q1 2015," Vision Mobile, p. 13.

developers, 56 percent had recently worked on Android as well."[159] It would have been as easy for an applications developer who had written an iPhone application in Objective C to port that application to C/C++ for use in Android as it was to port it to or write it from scratch in the Java programming language.[160]

107. Programmers familiar with one language typically find it relatively easy to pick up other languages.[161]

108. There are "translation" tools such as Mono/Xamarin that allow programmers to write an application in one language (say, Java) and then have the code translated to be used on a platform that uses a different language.[162] Such tools mean that a developer unfamiliar with the language used by a platform can still write applications for that platform and thereby decreases the importance of familiarity with any one language.

109. The history behind Google's choice to use Java as an applications programming language demonstrates that the advantages of Java over other languages was minimal. For example, Google witnesses testified regarding the debates as to which programming language to use.[163] Other languages that were considered included Python, JavaScript, Lua, and C++.[164]

---

[159] "Making Sense of a Fragmented World: Mobile Developer Economics 2010 and Beyond, Insights and Analysis from the Definitive Mobile Developer Survey Plus Benchmarks on the Platform Development Experience," VisionMobile Ltd, July 2010, pp. 11-13.

[160] Interviews of Tim Bray, Brian Swetland.

[161] Ghuloum 12/9/2015 Dep., p. 96:5-22. For an example, see Meier 12/11/2015 Dep., pp. 30:21-31:7.

[162] Ghuloum 12/9/2015 Dep., pp. 64:4-17, 74:13-24, 97:5-12; "Xamarin,"xamarin.com; www.mono-project.com.

[163] Trial Tr. 1597:13-23 (Rubin).

[164] Trial Tr. 1597:18-1598:14 (Rubin); Trial Tr. 1812:24-1813:13 (Bornstein); Rubin 4/5/2011 Dep., pp. 19:13-24:5. Mr. Ghuloum testified that to switch out the 37 APIs at issue today and replace them with C or C++ "wouldn't take much time at all," to replace them with RenderScript would take a single engineer six to 12 months, and to replace them with JavaScript would take substantially less than for RenderScript (Ghuloum 12/9/2015 Dep., pp. 74:13-76:6, 86:1-3). Moreover, the work would be scalable (meaning that the work could be divided among

110. There are disadvantages to Java relative to other programming languages. For example, Android apps written in Java, if run on the Dalvik virtual machine, run slowly relative to if they were written in C/C++.[165] Indeed, Google has received complaints from certain developers regarding Google's choice of the Java programming language because it is slow.[166] For that reason, Google offers C++ as a secondary applications programming language through the NDK. Most games, for example, are written in C++ because of the performance benefits inherent in avoiding the use of the virtual machine. In addition, app development can be more expensive for Android than for iOS in part due to Java programs requiring more code than Objective-C or Swift programs written to achieve a similar goal.[167]

111. Another disadvantage of the Java programming language that would have been avoided had C++ or Objective C been chosen as the applications programming language is that some Java ME developers have done a poor job porting their applications to Android, and this has had a negative externality on Android.[168] Similarly, some Java SE developers have had

---

engineers, reducing the total project time) (Ghuloum 12/9/2015 Dep., p. 74:13-24). Thus, at the time Google made its initial choice of applications programming language, there would have been no significant increase in the Android development time had it chosen to use C++ or JavaScript instead of Java as the applications programming language.

[165] Trial Tr. 1598:22-1599:4 (Rubin); Ghuloum 12/9/2015 Dep., p. 68:2-9.

[166] Interviews of Tim Bray, Brian Swetland.

[167] "Android Development is 30% More Expensive Than iOS," Infinum, October 27, 2015, https://infinum.co/the-capsized-eight/articles/android-development-is-30-percent-more-expensive-than-ios (on average, for Android an app requires 40% more code because Java "is simply a more verbose language than Objective-C or Swift…More often than not, having to write more code means working longer and more potential bugs.")

[168] Interview of Tim Bray. This demonstrates that the "write once, run anywhere" promise of the Java programming language is not fulfilled in practice with respect to the Java ME platform. See also "Making Sense of a Fragmented World: Mobile Developer Economics 2010 and Beyond, Insights and Analysis from the Definitive Mobile Developer Survey Plus Benchmarks on the Platform Development Experience," VisionMobile Ltd, July 2010, p. 14 (based on a survey of developers, "[o]ne could say that the vast majority of Java ME developers have lost faith in the write-once-run-anywhere vision").

trouble adapting to the mobile environment, again imposing a negative externality on Android.[169] In general, young programmers creating the most interesting applications do not like the Java programming language because it is considered obsolete.[170]

112. The disadvantages to Java have contributed to the fact that many applications available on both iOS and Android were launched first on iOS. According to a 2010 Nielsen study, Facebook, Pandora, and the Weather Channel are among the most popular apps (as measured by number of recent downloads) on both Android and iPhone.[171] Each of these apps was launched on iPhone before it was launched on Android. Facebook was launched on iOS in July 2008, as compared to July 2009 on Android; Pandora was launched on iOS in July 2008, as compared to September 2009 on Android; Weather Channel was launched on iOS and Android simultaneously.[172] Similarly, among the list of "Top 30 Android Apps of All Time" as determined

[169] Interview of Tim Bray.

[170] Interview of Tim Bray.

[171] "The State of Mobile Apps," Nielsen, June 1, 2010, http://www.nielsen.com/us/en/insights/news/2010/the-state-of-mobile-apps.html.

[172] "Facebook for iPhone Application Launches," Social Times, July 10, 2008, http://www.adweek.com/socialtimes/facebook-for-iphone-application-launches/212245; "The Weather Channel Tops 10 Million iPhone App Downloads," The Weather Company, April 28, 2010, http://www.theweathercompany.com/newsroom/2014/08/19/weather-channel-tops-10-million-iphone-app-downloads; "The Weather Channel App for Android Launches Major Updates," Lost Remote, February 28, 2013, http://www.adweek.com/lostremote/the-weather-channel-app-for-android-launches-major-updates/37856; "Happy Birthday to the Pandora App," Pandora Blog, July 10, 2013, http://blog.pandora.com/2013/07/10/happy-birthday-to-the-pandora-app/; "Pandora for Android," Pandora Blog, September 9, 2009, http://blog.pandora.com/2009/09/09/pandora_for_and/ .

by a 2010 TechCrunch article, 15 are non-Google apps that are also on iPhone.[173] Eleven of the 15 apps were launched on iPhone before they were launched on Android.[174]

113. The fact that developers tended to write their apps for iOS before Android demonstrates two points. First, these developers were not deterred in developing their apps for iPhone despite having to write the apps in Objective C, which was historically a little-used programming language and remains relatively unpopular today despite Apple's adoption of it

[173] The other 15 apps are Google-developed apps, utilities that are specific to the Android OS (such as an anti-virus app), or games exclusive to Android. "Top 30 Android Apps of All Time," TechCrunch, October 30, 2010, http://techcrunch.com/2010/10/30/top-30-android-apps/.

[174] "Kindle for Mac Now Finally Available," Engadget, March 18, 2010, http://www.engadget.com/2010/03/18/kindle-for-mac-now-finally-available/; "Amazon Releases Kindle App for Android Phones," New York Times, June 28, 2010, http://bits.blogs.nytimes.com/2010/06/28/amazon-kindle-app-now-available-for-android/?pagemode=print; "Evernote News," Evernote, July 10, 2008, https://evernote.com/corp/news/pr/2008-07-10.php; "Evernote for Android: It's Here!" Evernote, December 16, 2009, https://blog.evernote.com/blog/2009/12/16/evernote-for-android-its-here/; "Facebook for iPhone Application Launches," Social Times, July 10, 2008, http://www.adweek.com/socialtimes/facebook-for-iphone-application-launches/212245; "Facebook Launches Official Google Android Application," Mashable, September 8, 2009, http://mashable.com/2009/09/08/facebook-android/#MUlofPT.rsqz; "Becomes No. 1 iTunes Finance Application Within 24 Hours ," Mint, December 22, 2008, https://www.mint.com/press/mint-introduces-free-iphone-application; "Mint Comes to Android," Readwrite, May 3, 2010, http://readwrite.com/2010/05/03/mint_comes_to_android; "Facebook for iPhone Application Launches," Social Times, July 10, 2008, http://www.adweek.com/socialtimes/facebook-for-iphone-application-launches/212245; "Sing for Search Results with iPhone App," CNET, July 11, 2008, http://www.cnet.com/news/sing-for-search-results-with-iphone-app/; "Dropbox for iPhones Is Out (and Awesome)," Mashable, September 29, 2009, http://mashable.com/2009/09/29/dropbox-iphone/#to8U3hEnrZqW; "The Dropbox Android App," Dropbox Blog, March 22, 2010, https://blogs.dropbox.com/dropbox/2010/03/the-dropbox-android-app-2/; "First Look: LogMeIn Ignition," Engadget, January 19, 2009, http://www.engadget.com/2009/01/19/first-look-logmein-ignition/; "LogMeIn Ignition Launches Today, $29.99 in the Android Market," Phandroid, July 14, 2010, http://phandroid.com/2010/07/14/logmein-ignition-launches-today-29-99-in-the-android-market/; "Fruit Ninja," Gamespot, April 20, 2010, http://www.gamespot.com/fruit-ninja/; "Fruit Ninja now available on Android!," Half Brick, September 17, 2010, http://halfbrick.com/fruit-ninja/fruit-ninja-on-android/; "HomeRun Battle 3D," IGN, June 17, 2009, http://www.ign.com/games/home-run-battle-3d/iphone-14356201; "Homerun Battle 3D goes Goid, er… Droid," TouchMyApps, January 7, 2010, http://www.touchmyapps.com/2010/01/07/homerun-battle-3d-goes-goid-er-droid/; "The Best iPhone Apps of 2009," Tech Crunch, December 27, 2009, http://techcrunch.com/2009/12/27/best-iphone-apps-2009-appvee/; "Zenonia and other Top Android Games of the Week," App Olicious, August 16, 2010, http://www.appolicious.com/articles/2794-zenonia-and-other-top-android-games-of-the-week; "Rovio's 'Angry Birds 2' to launch on Apple's iOS July 30," Apple Insider, July 16, 2015, http://appleinsider.com/articles/15/07/16/rovios-angry-birds-2-to-launch-on-apples-ios-july-30; "Angry Birds" Now Available on Android for Free," Mashable, October 15, 2010, http://mashable.com/2010/10/15/angry-birds-android-2/#EmCWRAtkyqqF.

for use with iOS devices. In fact, some developers prefer Objective C to Java given its greater efficiency and more up-to-date libraries.[175] Second, each company delayed developing its app for Android despite being able to write the app in Java, most likely to wait for Android to demonstrate sufficient user growth to show that Android offered a market opportunity worth exploiting by developers (a topic discussed in greater detail below). This demonstrates that a platform's revenue potential is a much more important driver of developer choice to develop for the platform than is its applications programming language.[176]

114. If Oracle is correct that Google's use of Java as an applications programming language in Android made app development easier due to developer familiarity with Java, we should observe that, of the developers that dual-homed (i.e., offer their app on both iPhone and Android), those that developed their Android apps in Java exhibited a shorter lag between the iPhone and Android launches of their apps than those that developed their Android apps in C++ (using the NDK). This hypothesis can be tested by running a regression of a developer's iPhone/Android launch lag on an indicator variable for whether the Android app was written in Java. Effects of relative installed base can be controlled for by also including indicator variable for time of the app's launch as explanatory variables.

115. From App Annie data, I identified apps appearing on Top 100 monthly download lists during the 2012-2013 period that multi-homed on both the Android and iPhone platforms.

---

[175] "Reservoir Devs: Why Apps Really Hit iOS Before Android," December 17, 2015, http://thenextweb.com/dd/2015/12/17/reservoir-devs-why-apps-really-hit-ios-before-android/#gref. This article summarized responses of developers who were asked about their platform choices. One developer stated, "There are so many incredible third-party libraries made with Swift and Objective-C now that in my mind developing for Android is like being in a fight with one hand tied behind your back."

[176] Ghuloum 12/9/2015 Dep., pp. 148:12-149:3.

For each such app, I calculated the difference in time between the Android launch and the iPhone launch. I determined whether the Android version of each app was written in Java or C++. Then, I performed the regression described above. I find that the coefficient on the Java indicator variable is positive, contrary to what would be expected if Oracle's claim is correct.[177] That is, contrary to what Oracle's claim would predict, the lag in launch time between iPhone and Android is estimated to be larger when the Android app is written in Java. However, the difference is imprecisely estimated so that the hypothesis of equal lag times between Java and C++ is not rejected. Thus, the conclusion I reach is that the data are not consistent with the claim that Java makes Android app development easier.

b. A Substantial Fraction of the Most Popular Mobile Apps Are Not Written in the Java Programming Language

116. In contrast to Google, Apple offers a proprietary smartphone operating system and maintains complete control of the smartphone hardware. Apple uses Objective C and Swift as its applications programming languages. Thus, essentially the entire universe of iPhone applications are not written in Java. That, together with the fact that Android games are generally written in C++, implies that a substantial percentage of the most popular mobile apps are not written in Java.

c. The Apple Experience Demonstrates That Success of a Smartphone Platform Does Not Require Use of Java as an Applications Programming Language

117. At the time Apple chose Objective C as its applications programming language, Objective C was a substantially less popular programming language than Java, JavaScript,

[177] Exhibit 2c.

Python, and C/C++.[178] Even after the iPhone's success was established and a large number of applications had been developed for the iPhone platform, Objective C was still ranked well below other "top tier" languages.[179] Yet, despite the lower levels of familiarity with Objective C, there has been an explosion in applications developed for the iPhone.[180] This shows that developers are willing and able to respond to market opportunities by adapting and learning to program in previously unfamiliar languages.

118. Even after Android surpassed iPhone in terms of shipments and user base, it was still accepted as common wisdom in the industry that "developers should write for iOS (Apple) first and then Android (Google) second."[181] This is not consistent with Android having a substantial advantage over iOS due to using Java as an applications programming language.

119. In 2014, Apple introduced an entirely new programming language called Swift as an applications programming language for iOS devices, with the ultimate intention of replacing Objective C in that role. Despite being entirely new, Swift has quickly gained acceptance and iOS developers have switched over from Objective C in great numbers, because of its adoption by Apple, an established and successful smartphone-platform owner, and the size and relative certainty of the market opportunity offered to developers willing and able to learn and program

[178] According to the TIOBE index cited by Oracle's experts, Objective C was ranked low in popularity. http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html. This is still the case. A 2015 survey of developers (not restricted to mobile app developers) found that only 36% of respondents were moderately or highly conversant with Objective C as compared to 68% for Java. Developer Insights Report, IDC, August 2015, p. 26. This study notes that Java number has benefited from Android, so the two numbers would have been closer at the time of Android's launch. Id.

[179] Id.

[180] "Mobile Developer Economics 2010 and Beyond," VisionMobile, July 2010, p. 10.

[181] T. Bresnahan, J. Davis, and P.L. Yin, "Economic Value Creation in Mobile Applications," in The Changing Frontier: Rethinking Science and Innovation Policy (A. Jaffe and B. Jones, eds., 2015) (hereafter "Bresnahan, et al. (2015)), at location 5838 (Kindle version).

in Swift.[182] This further underscores the minor importance of the specific language that is chosen to be the applications programming language for a smartphone platform.

120. Apple has announced that it will make Swift open source.[183] This may be a response to competitors such as Microsoft, which in 2014 open sourced its .NET framework that allowed programmers to develop code that would run on iOS.[184] In addition, a company called Xamarin offers a framework that allows a developer to use C# to write programs that are then converted by the framework to run on Android and iOS devices (as well as Microsoft devices).[185]

d. The BlackBerry Experience Demonstrates That Success of a Smartphone Platform Is Not Ensured By Use of Java as an Applications Programming Language

121. In contrast to Google, until recently BlackBerry offered a proprietary smartphone operating system and maintained complete control of the hardware. BlackBerry uses Java ME for its platform. As of early 2010, a Nielsen study found that the most downloaded apps on Blackberry were Facebook, Google Maps, Weather Channel, ESPN, and Pandora—very similar to the most downloaded apps on Android (Google Maps, Facebook, Weather Channel, Pandora, and Google Search).[186]

---

[182] "The RedMonk Programming Language Rankings: June 2015," RedMonk, June 2015, http://redmonk.com/sogrady/2015/07/01/language-rankings-6-15/; "Apple's Biggest Breakthrough That Almost No One Knows About," Bloomberg, June 4, 2015, http://www.bloomberg.com/news/articles/2015-06-04/apple-s-big-breakthrough-that-almost-no-one-knows-about-swift-code. "State of the Developer Nation Q1 2015," VisionMobile, Q1 2015, p. 5.

[183] "Swift 2.0," Apple, June 8, 2015, https://developer.apple.com/swift/blog/?id=29.

[184] "Open Sourcing Is No Longer Optional, Not Even For Apple," Wired, June 9, 2015, http://www.wired.com/2015/06/open-sourcing-no-longer-optional-not-even-apple/.

[185] Id.

[186] "The State of Mobile Apps," Nielsen, June 1, 2010.

122. Yet, despite having Java as an applications programming language and despite offering users many of the same most popular apps as Android, BlackBerry quickly lost out to the iPhone and Android handsets. This demonstrates the minor role that the particular choice of applications programming language plays in the success of a platform as compared to features of the hardware, features of the operating system itself, and strategic choices of the vendor.

e. A Small Number of Apps Are Responsible for a Large Share of the Overall Value of Apps to Users

123. Even assuming that the contribution of the alleged infringement was to allow for more Android apps at any point in time, it is likely that these "incremental" apps are of marginal and of low value to consumers.

124. While there are many Android apps currently, the large majority of downloads and usage are accounted for by a relatively small number of apps. According to a 2014 Nielsen study, the average consumer used on average 26.8 apps per month which was only a slight increase from 23.2 apps per month in 2011.[187] A 2015 Forrester study found that smartphone users spend the majority of their time on five apps, and that a small number of companies (one

[187] "Smartphones: So Many Apps, So Much Time," Nielsen, July 1, 2014, http://www.nielsen.com/us/en/insights/news/2014/smartphones-so-many-apps--so-much-time.html.

of which is Google) provide the apps that users spend most of their time on.[188] According to a 2015 Nielsen study, in 2014 70% of total app usage came from the top 200 apps.[189]

125. A 2014 comScore study found that 75% of an individual's app usage is accounted for by only four apps.[190] A recent economic study analyzed comScore data and found that the distribution of app demand is highly skewed, i.e., a relatively small number of apps account for a large majority of app demand.[191]

126. I have analyzed the comScore data and found that it supports these conclusions. comScore tracks the app usage of a sample of smartphone users.[192] For each month during the January to March 2013 and January to March 2015 periods, I calculated the percentage of overall app usage that is accounted for by the top 10, 50, 100, and 200 apps in each month.[193] The top 10 apps account for about 50% of usage and the top 200 apps account for 80% or more of usage.

---

[188] "Consumers Spend 85% of Time on Smartphones in Apps, But Only 5 Apps See Heavy Use," Tech Crunch, June 22, 2015, http://techcrunch.com/2015/06/22/consumers-spend-85-of-time-on-smartphones-in-apps-but-only-5-apps-see-heavy-use/#.rbf9gcs:5bJy.

[189] "So Many Apps, So Much More Time for Entertainment," Nielsen, June 11, 2015, http://www.nielsen.com/us/en/insights/news/2015/so-many-apps-so-much-more-time-for-entertainment.html.

[190] "The State of Mobile Industry", comScore, September 17, 2014, p. 12. The four apps will differ to some extent across users.

[191] Bresnahan, et al. (2015), at location 6117 (Kindle version). Another study using the same data concluded similarly that "[v]ery few apps are providing most of the value for either platform." "Platform Choice by Mobile App Developers," T. Bresnahan, et al., Stanford University, May 29, 2014 (henceforth "Bresnahan, et al. (2014)), p. 12.

[192] The Mobile Metrix of comScore provides data on monthly app usage behaviors, such as total minutes spent, unique visitors, and average daily visitors by app, collected from an online survey on a nationally representative sample of over 10,000 mobile subscribers aged 18 or higher each month in the United States. "comScore Mobile Metrix Methodology," comScore Inc., June 2013.

[193] Exhibit 2d.

127. The need for only a relatively small number of apps was particularly true in mobile platforms' early days. A recent economic study concluded that

> [G]et[ting] over the hump into viability…was achieved by the supply of a modest number of influential apps, which, taken together, were sufficient to attract a significant body of demanders. These included a media store, especially for music (iTunes), an app for accessing the web (mobile browser), an informative map, and, after a brief interval, some games. Together with some ease-of-use improvements over existing phones, and some economies of purse and pocket space (phone and music in one device), the supply of these "killer apps" led to an initial expansion in user demand for smartphones that could run apps. This created an enormous market for new apps.[194]

In the case of Android, Google supplied the mobile browser and interactive map; Amazon provided a music store app at Android's launch[195] and Google offered a music store starting in May 2011[196]; and games are generally written in C++, not Java. Thus, Java was not essential, or even important, for the provision of the "killer apps" needed to get Android going.

128. This same economic study also concluded that "the majority of apps in the raw count are marginal apps and have not been downloaded or used by customers."[197]

129. The willingness of developers of apps with substantial downloads and usage to develop those apps was unlikely to have been driven by Google's use of the "familiar" allegedly infringing material. Instead, it would have been developers of minor apps whose participation decision potentially was affected by the decreased "costs" (psychic or otherwise) associated

---

[194] Bresnahan, et al. (2015), at location 5883 (Kindle version, footnote omitted).

[195] "Top Ten Android Launch Apps," Tech Crunch, http://techcrunch.com/2008/10/22/top-ten-android-launch-apps/.

[196] "Introducing Google Play: All Your Entertainment, Anywhere You Go," Google Official Blog, https://googleblog.blogspot.com/2012/03/introducing-google-play-all-your.html.

[197] Bresnahan, et al. (2015), at location 5883 (Kindle version).

with the familiar allegedly infringing material. But, because the apps of these developers are of little value to users, they contributed little to the value of the Android platform.

f. Many of the Apps With Most Value to Consumers Are Provided By Companies That Would Have Had Strong Incentives and Ability to Provide Those Apps for the Android Platform Regardless of the Applications Programming Language

130. Facebook is by far the overall most downloaded Android app (and Facebook Messenger is another highly downloaded app). Facebook is a substantial company with tremendous resources and strong incentives to provide mobile apps on any mobile platform with a meaningful user base or the promise of acquiring one. Accordingly, Facebook would have been willing to incur any additional expenses that would have been required to code its Android app in another programming language. In other words, there is no reason to believe that the alleged infringement, or even the use of the Java language generally, is the reason why Facebook chose to offer an Android app. Indeed, as discussed below, Facebook "multi-homes," meaning it provides apps on other platforms, such as iPhone, in addition to Android. Thus, Facebook has demonstrated the ability to provide mobile apps written in other programming languages, which presumably use the APIs associated with those languages.[198]

131. Many of the other most popular apps were also developed by companies with significant resources and strong incentives to offer apps on Android regardless of the applications programming language. As a 2014 comScore report noted "[t]he largest digital

[198] Facebook rebuilt its iOS and Android apps in native code in 2012. See "Under the Hood: Rebuilding Facebook for Android," Facebook, December 13, 2012, https://www.facebook.com/notes/facebook-engineering/under-the-hood-rebuilding-facebook-for-android/10151189598933920/; "Under the hood: Rebuilding Facebook for iOS," Facebook, August 23, 2012, https://www.facebook.com/notes/facebook-engineering/under-the-hood-rebuilding-facebook-for-ios/10151036091753920/.

media brands ([Facebook, Google, Apple, Yahoo, Amazon, eBay]) make up 16 of the 25 most used apps."[199]

132. A recent economic study notes that "[costs of offering an app] appear to be substantially lower for corporate apps than for apps from entrepreneurs."[200] That is, a corporate entity faces relatively low costs to offer an app (this is due to lower cost of "marketing" the app that result from the entity having a built-in user base for its app[201]). Yet, such corporate entities supply the most popular apps.[202]

133. This study also notes that many firms that develop apps for Android also develop apps for iOS.[203] Thus, these firms have the capability to develop apps for both platforms. In that environment, Android's use of Java provided little advantage. Notably, it is "the most important apps" that are developed for both platforms and "the most demand apps tend to be from established firms."[204] Thus, the most important apps account for most user demand and are developed by companies that would not have been deterred had Java not been Android's applications programming language.

134. I have used the App Annie data to identify the top 100 developers by downloads of their Android applications by year from 2012 through 2015.[205] The large majority of these

---

[199] "The State of Mobile Industry," comScore, September 17, 2014, p. 15.

[200] Bresnahan, et al. (2015), at location 5838 (Kindle version).

[201] Id. at location 6701 (Kindle version).

[202] Id. at locations 6652, 6701 (Kindle version).

[203] Id. at location 6688 (Kindle version). Bresnahan, et al. (2014) notes that developers say that "porting [between Android and iOS] is not extremely difficult technically" (p. 8). This suggests that Java does not provide any particular advantages over Objective C from the point of view of developers.

[204] Bresnahan, et al. (2014), pp. 1, 4.

[205] Exhibits 2e and 2f.

developers are familiar corporate entities with substantial resources and strong incentives to supply apps on Android.

g. Google Itself Is the Developer of Many of the Most Popular and Most Used Android Apps

135. Google is recognized as one of the top developers of Android apps.[206] Google provides apps within a large number of categories, including travel, music, communications, social networking, tools, books and reference, productivity, shopping, news and magazines, entertainment, health and fitness, and finance.[207]

136. According to a 2015 Pew Research survey of smartphone owners, the most widely used functionalities among smartphone users are text messaging, voice and video calls, internet browsing, and email, and text messaging and email are also the most frequently used functionalities.[208] Android and GMS provide the entirety of these functionalities for Android users; no third party app is required.

137. A 2011 study of Android mobile app usage surveyed mobile phone users to collect data on their app usage, categorizing apps into categories such as "Tools," "Browser," and "Travel."[209] Android and GMS provide functionality within most of the categories defined by the study. For example, Chrome is in the Browser category and Google Maps is in the Travel category.

[206] "Top Android Developers," Android APK.

[207] See Google Play.

[208] "U.S. Smartphone Use in 2015," Pew Research Center, April 1, 2015, p. 33.

[209] M. Bohmer, et al., "Falling Asleep with Angry Birds, Facebook and Kindle – A Large Scale Study on Mobile Application Usage," MobileHCI 2011, September 2, 2011.

138. A 2010 Nielsen report noted that the most popular apps on Android (measured by downloads) were Google Maps, Facebook, Weather Channel, Pandora, and Google Search. Google was responsible for two of these five apps. A 2015 comScore report identified the top 10 Android apps (measured by number of users) as Facebook, YouTube, Google Search, Facebook Messenger, Google Play, Google Maps, Gmail, Pandora Radio, Instragram, and iTunes Radio/Cloud. .[210] Google is responsible for six of these ten apps. As comScore notes, "[w]ith the exception of Facebook and Pandora apps, Google-created apps dominate usage on Android Phones."[211]

139. Google provides a large fraction of the relatively small number of apps that are highly used. Given Google's commitment to the success of the Android platform, its decision to develop its apps for Android was not in any way dependent on its use of the allegedly infringing material.[212] For example, while I understand that the GMS apps are written in the Java programming language, Google would have faced no constraints in programming them in a different language. Indeed, Google has in fact provided most of these same apps written in Objective C for use on the iPhone. Thus, it would be incorrect to attribute to the allegedly infringing material any portion of the contribution to Android's success that was provided by the Google apps.

---

[210] "comScore Reports April 2015 U.S. Smartphone Subscriber Market Share," comScore June 5, 2015.

[211] G. Fulgoni, "The State of Mobile," comScore, September 17, 2014 at 16.

[212] It is also the case, not surprisingly, that Google employs programmers with a wide range of language experience. Indeed, the Java programming language is only the second or third most popular language within Google (Holzle 12/24/2015 Dep., pp. 133:19-134:14).

h. The Contribution of the Alleged Infringement Was To Allow Google to Save on Training or Development Costs, Which Are Limited

140. The primary value to Google of using the "familiar" allegedly infringing material was to avoid the costs that it would have incurred had it not used this material. For example, to overcome any adverse effects on Android's success of supporting only C/C++ as the applications programming language, Google could have provided training to those Android developers that were familiar with Java, but not C++. Alternatively, Google could have provided financial incentives to developers to develop Android apps. Such payments would have encouraged Android developers not familiar with C++ to learn C++ or outsource the C++ programming to others and would have induced entry by additional C++ programmers into Android app development.

141. This characterization of the contribution of the alleged infringement is consistent with the evidence in the record. Google witnesses testified that the advantage of using Java as an applications programming language (which, again, was greater than the advantage of just using the allegedly infringing material) was that app programmers were familiar with this language and thus would not require retraining.[213] Even Oracle's counsel argued that "[Google's] commercial objective" when allegedly infringing was "so [applications programmers] wouldn't have to be retrained."[214]

142. It is important to note that familiarity with the allegedly infringing material is only a minor benefit to a developer who was learning how to use the Android platform. This is

[213] Trial Tr. 1587:10-1588:2 (Rubin); Bloch 7/8/2011 Dep., p. 55:5-18; Meier 12/11/2015 Dep., p. 113:9-19; Ghuloum 12/9/2015 Dep., p. 11:1-19.

[214] Trial Tr. 2496:7-20 (Jacob).

because there are a large number of other APIs and functionalities that Android offers and that a developer must understand to use the platform effectively. As Mr. Meier of Google testified:

> Q: Would you consider that it would be easier for a developer who is already familiar with the Java APIs to learn Android as opposed to developer who wrote applications only for iOS?
>
> A: The Java APIs specifically? No. I wouldn't say it would have been a significant advantage.
>
> Q: And why not?
>
> A: For the most part, the hard part of learning a new platform is all the more complicated things which go on top of it, so the Android-specific stuff is the hard part to learn, how to use the location-based services and content providers and all of the technical APIs around Android. Most of the – most of that is going to be specific to different platforms…[215]

143. Given that the contribution of allegedly infringing material to a developer's ease in writing Android applications is limited, it is not surprising that Google has no requirement that "developer advocates" (i.e., Google personnel that work with developers to promote the development of Android apps) know the Java programming language specifically.[216]

144. At the same time, learning a new programming language is not difficult for an experienced programmer because of the commonalities in basic programming concepts across languages. As Mr. Meier testified, "underlying language syntax, and some of the basic APIs are fairly consistent across programming languages and, if not, are fairly easy to sort of learn quickly."[217]

---

[215] Meier 12/11/2015 Dep., p. 104:4-18.

[216] Rutledge 12/9/2015 Dep., p. 30:7-24.

[217] Meier 12/11/2015 Dep., p. 104:18-21, 108:18-19 ("the core libraries for most languages will be very similar").

145. For these reasons, the costs required to train Java programmers (who did not already know C++, Python, or whichever language Google might have chosen) to use another language would be limited in magnitude. For example, the cost of a C++ course is approximately $700.[218] Google could have paid for training for any developer who was not already familiar with C++. The use of the allegedly infringing material at most saved Google from having to incur these expenses.

146. Alternatively, to the extent that developing an app in a language other than Java was more costly to some developers (due, e.g., to lack of familiarity), Google could have paid developers to develop apps (or outsource the app for development) in that language. Google already devotes a team to work with Android developers[219], and prior to Android's launch Google ran a contest for developing Android apps and paid out awards to the winners.[220] Paying developers to develop apps is also something Microsoft has actually done.[221]

---

[218] See, e.g., "C/C++ Programming I: Fundamental Programming Concepts," US San Diego Extension, http://extension.ucsd.edu/studyarea/index.cfm?vCourse=CSE-40475; "C++ Programming," UC Berkeley Extension, http://extension.berkeley.edu/search/publicCourseSearchDetails.do?method=load&courseId=40931.

[219] Meier 12/11/2015 Dep., p. 14: 7-11.

[220] "Fifty Android Developers Get $25,000 Each: The List," TechCrunch, May 13, 2008, http://techcrunch.com/2008/05/13/fifty-android-developers-get-25000-each-the-list/; "The Best of Android: Final Challenge Winners Announced," TechCrunch, August 28, 2008, http://techcrunch.com/2008/08/28/final-winners-of-android-challenge-announced/.

[221] See "Microsoft Pays Mobile App Developers to Catch Apple," Bloomberg, July 14, 2010, http://www.bloomberg.com/news/articles/2010-07-14/microsoft-pays-developers-to-build-mobile-apps-to-help-catch-up-with-apple (Microsoft used revenue guarantees and cash payments); "Microsoft Still Paying Developers to Create Windows Phone Apps," CNET, April 6, 2012, http://www.cnet.com/news/microsoft-still-paying-developers-to-create-windows-phone-apps/ (Microsoft made payments from $60,000 to $600,000 to developers). Using payments to developers to offset development costs is also something Google contemplated, but ultimately decided not to do (aside from the contest mentioned in the text) because the payments involved would not convince enough developers who would not otherwise develop an app to do so (Meier 12/11/2015 Dep., pp. 70:1-72:20). However, in this case, the question is the incremental cost to a developer who actually did decide to develop an app in Java of having to use an alternative language such as C++. The entire initial development cost would be a conservative measure of the relevant incremental cost.

147. Mr. Meier of Google estimated that apps cost $75,000 to $100,000 to develop.[222] Other estimates are as low as $22,673.[223]

i. Sun Open-Sourced Java SE, Including the 37 API Packages, Demonstrating That the Value of the Allegedly Infringing Material Was Limited

148. In November 2006, Sun announced that it would open-source the Java SE platform.[224] In May 2007, Sun provided for an open-source Java SE implementation called OpenJDK.[225] Various companies, including Google, have contributed source code to OpenJDK. The OpenJDK implementation is now the "reference" implementation for Java SE.[226] The OpenJDK includes the 37 API packages at issue in this case.

149. Use of the OpenJDK code for the 37 API packages at issue is governed by a royalty-free GPU GPL license with a Classpath linking exception.[227] This exception allows a

---

Once the Android user base was sufficiently large, the payments would no longer be necessary to induce developers to develop apps.

[222] Meier 12/11/2015 Dep., p. 70:16-19.

[223] "Average Cost to Develop a Mobile Phone App in 2012, By Operating System," Statista, 2012, http://www.statista.com/statistics/256541/average-cost-to-develop-an-app-by-os/.

[224] "Sun Fulfills Promise of Open and Free Java Technology and Releases Java SE Platform to OpenJDK Community; Advances OpenJDK Project with New Code, NetBeans Integration, Governance Board and Availability of Compatibility Tests," PR Newswire, May 8, 2007.

[225] "Sun Announces Open Source Community Innovation Awards Program; Multi-Year Program Expected to Payout Millions to Foster Global Community-Based Innovation," Business Wire, December 5, 2007; "Sun Fulfills Promise of Open and Free Java Technology and Releases Java SE Platform to OpenJDK Community; Advances OpenJDK Project with New Code, NetBeans Integration, Governance Board and Availability of Compatibility Tests;" PR Newswire; May 8, 2007; "IcedTea: The First 100% Compliant Open-Source Java," InfoQ News, June 21, 2008, available at http://www.infoq.com/news/2008/06/icedtea_tck; "Open source Java could result in port to iPhone," InfoWorld, November 9, 2007, http://www.infoworld.com/article/2650604/application-development/open-source-java-could-result-in-port-to-iphone.html.

[226] "Moving to OpenJDK as the official Java SE 7 Reference Implementation," Oracle blog, https://blogs.oracle.com/henrik/entry/moving_to_openjdk_as_the; "Java Platform, Standard Edition 8 Reference Implementations," Java.net, https://jdk8.java.net/java-se-8-ri/.

[227] "OpenJDK FAQ," Open JDK, December 18, 2010, available at http://openjdk.java.net/faq/.

developer to call the OpenJDK implementation within a larger application without having to open source the remainder of the application.[228]

150. The fact that Sun chose to open-source its Java SE reference implementation (i.e., provide a royalty-free implementation that could be used to build a platform like Android), including the allegedly copyrighted material, demonstrates that the contribution of the allegedly infringing material to the value (profit) of a platform like Android was not particularly large.[229] Otherwise, Sun would not have allowed for the use of OpenJDK, including the allegedly copyrighted material, for free. As discussed above, the limited value of the allegedly infringing material was due to the existence of alternatives to its use (i.e., use a different applications programming language) and to Java's decline in popularity among programmers, a trend that Android eventually became responsible for reversing.

3. Reasons for the Growth in the Number of Android Apps

151. While Oracle and its experts claim that Google's use of the allegedly infringing material led to an increase in the number of Android apps and thereby an increase in the number of Android users, the economic evidence demonstrates that other factors were responsible for the growth in Android users and that the growth in users drove the growth in apps to a much greater degree than the other way around. Above, I have discussed the economic evidence that supports the conclusion that the contribution of the allegedly infringing material to the value of the Android platform was small in both relative and absolute senses. Here, I specifically address the relationship between Android apps and Android users.

[228] "GNU Classpath," http://www.gnu.org/software/classpath/license.html.

[229] As Mr. Rubin, head of Android at Google at the time, noted in a November 12, 2006 email, the Sun decision to open source Java "has got to bring the price down" (TX 154).

a. Economics of Multi-Sided Platforms

152. A multi-sided platform is a vehicle for connecting members of one group with members of another group.[230] For an operating system like Android, the primary groups of interest are device OEMs, users, and applications developers. In principle, increased attractiveness of any one side of a multi-sided platform can promote increases in the number of participants on other sides of the platform. For example, as the number of users of an operating system increases, the platform becomes more attractive to applications developers, all else equal. The platform owner may have various "levers" it can pull to increase attractiveness to users on a given side of the platform, which may then increase the attractiveness to users on other sides of the platform. For example, dating platforms famously pay premiums to women to use the platform, while charging men for their use. In the case of a given platform, therefore, the degree to which increases in users on one side of the platform caused increases in users on the other sides is an empirical question that depends on the facts of the situation.

153. In the case of Android, a review of the empirical evidence demonstrates that the growth in users (which in turn was a result of a wide variety of devices provided by OEMs, improved functionality of the Android OS, and the promotional efforts of Google, OEMs, and carriers, among other things) primarily drove the growth in apps, rather than the other way around, particularly in the period immediately following the Android introduction.

[230] See, e.g., J.C. Rochet and J. Tirole, "Platform competition in two-sided markets," Journal of the European Economic Association (2003), pp. 990-1029; J.C. Rochet and J. Tirole, "Two-sided markets: a progress report," RAND Journal of Economics (2006), pp. 645-667.

b. Drivers of App Developers' Decisions Regarding Platforms

154. An important consideration for an app developer deciding whether to develop an app for a given platform is the profit that the app is expected to generate.[231]

155. An app may generate revenues from fees paid by users of the app or from advertisement placements. Costs can include the costs of development, the costs of placement in an app store, and marketing costs, among others.[232]

156. As a recent economic study has noted, "platform providers Apple and Google have lowered the costs of development and distribution of mobile applications," which has been a substantial cause of the "rapid growth in app development."[233] These effects of the platforms on the costs to app developers have nothing to do with the specific applications programming language used by the platforms.

157. The revenue generated by an app generally will increase with the number of users on the platform.[234] On the other hand, many of the developer's costs in bringing an app to market (such as the costs of software development and the costs of placement in an app store) are largely fixed with respect to the number of users on the platform. Thus, the developer's expected profits from developing the app for a given platform will generally

[231] "Five Reasons Why Google Android versus Apple iOS Market Share Numbers Don't Matter," Forbes, August 22, 2012, http://www.forbes.com/sites/darcytravlos/2012/08/22/five-reasons-why-google-android-versus-apple-ios-market-share-numbers-dont-matter/#a13782b39967

[232] "101 Ways to Make Money with Your iPhone, Android or Mobile App," Monetize Pros, September 17, 2013, http://monetizepros.com/features/101-ways-to-make-money-with-your-iphone-android-or-mobile-app/.

[233] Bresnahan, et al. (2015), at location 5795 (Kindle version).

[234] It will also depend on the price that can be charged, the percentage of handsets that can run the app, etc. See Meier 12/11/2015 Dep., pp. 45:4-46:24.

increase with the number of users.[235] This implies that a developer's decision as to whether to invest in the development of an app for a given platform will depend on the number of users of that platform, all else equal, and, aggregating over developers, that the number of apps generally will increase with the number of users.

158. However, it should be noted that a substantial portion (about 65%) of the most popular iPhone and Android apps are also on Microsoft Windows Phone, despite its relatively low user base (and the fact that it uses C# as the applications programming language).[236]

159. Another factor that may enter into an app developer's decision to develop for a given platform is the costs to place on app on the platform's app store. For example, Android charges a one-time fee of $25 to publish on Google Play, while Apple charges $99 to $299 annually depending on the type of account.[237]

160. The ease of the development process is another potential factor that can affect a developer's choice as to whether to develop an app for a given mobile OS. Ease of development can depend on the applications programming language, the availability of development environment tools, and the ability to do development on different computer

---

[235] See, e.g., Bresnahan, et al. (2015), at location 6527 (Kindle version) ("the most important determinant of developer behavior is the installed base"). See also Barr 12/9/2015 Dep., p. 133:8-12; Meier 12/11/2015 Dep., pp. 24:1-24.

[236] ActiveNick, "Top 100 Apps Availability on iOS, Android, Windows, Phone & Windows 8," Infragistics, August 8, 2013 (last update), http://www.infragistics.com/community/blogs/nick-landry/archive/2013/08/06/top-100-apps-availability-on-ios-android-windows-phone-amp-windows-8.aspx. As this publication notes, "Android had to reach a MUCH higher market share [than Windows Phone 8] before official brands started developing for the platform."

[237] "Android Market: Now available for users," Android Developers Blog, http://android-developers.blogspot.com/2008/10/android-market-now-available-for-users.html; "Choosing a Membership," Apple Developer Website, https://developer.apple.com/support/compare-memberships/.

systems.[238] With regard to the applications programming language, however, the cost of using an unfamiliar language is limited. As one study quoted one of its respondents with respect to Objective C versus Java, "Objective C is very different from any object-oriented language…we are used to such as Java. But from the moment that you break that barrier, it becomes much simpler to develop [using Objective C] than [with Java for] Android."[239]

161. Moreover, many factors go into how much time is required for a developer to learn to develop an application on a platform, and programming language is at best only a minor such factor. One developer survey found that Android required the least time to "master"; however, Java ME and the Blackberry platform required significantly greater time to master than Android despite also using the Java programming language (they also required significantly greater time to master than iPhone).[240]

c. Evidence From the Survey of App Developers

162. In connection with this litigation, Dr. Itamar Simonson of Stanford University has conducted a survey of applications developers, including developers who create applications

[238] "An Exploratory Study of the Adoption of Mobile Development Platforms by Software Engineers," M. Muller, et al., 1st International Conference on Mobile Software Engineering and Systems, 2014. This study compares Android and iOS on several dimensions that do not have to do with Java, such as "testability," "observability," "complexity," and "relative advantage." Java enters into "compatibility" (there are more Java programmers than Objective C programmers), but the study notes that this advantage is limited for the reason quoted in the text.

[239] Id. at p.3.

[240] "Mobile Developer Economics 2010 and Beyond," Vision Mobile Research, July 2010, pp. 31-32. This survey also found that the three most important factors in a developer's decision as to whether to develop on a given platform are large market penetration, revenue potential, and presence of an app store. Id. at p. 13.

for both the Android and iOS operating systems. The results of the survey are consistent with and confirm the conclusions of the above discussion, including:[241]

- The size of a mobile platform's user base and the expected profitability of developing for that user base are the most important considerations in developers' decisions to develop for a platform, while familiarity with the programming language is a minor consideration at best for only a small number of developers (and even then, a far less important consideration compared to the others).
- There is no reason to believe that developers would decline the opportunity to develop for a new mobile platform that showed market potential (for example, through a growing base of users) because of the programming language used by that platform.
- Many Android developers also develop for iOS, which suggests that many developers know and work using different programming language and different applications programming environments, demonstrating the flexibility to respond to market conditions and opportunities.
- Developers are confident that they could learn a new programming language, with many in fact first learning Objective C when they started developing for iOS and then learning or planning to learn Swift, the new iOS programming language recently introduced by Apple.

d. Growth in Android Apps Followed the Growth in the Number of Android Users, Not Vice-Versa

163. In 2009 and 2010, Android app development lagged behind iPhone app development, despite the fact that the Android user base was growing.[242] Developers were hesitant to initiate Android app development before there was a return that could justify the cost, and this break-even return depended in part on the existence of a sufficiently large user

[241] See generally Simonson 2/8/2016 Report.

[242] See, e.g., "App Makers Take Interest in Android," New York Times, October 25, 2010, http://www.nytimes.com/2010/10/25/technology/25android.html?_r=0 ("Even as Android sales surge – Google says it is now activating around 200,000 phones per day – the market for Android apps still seems anemic compared with that for Apple and its thriving App Store"); "The Rise of Android, A Deep Dive Analysis," Caris & Company, September 13, 2010.

base.[243] As a recent economic study concludes, "the rapid emergence of many demanders, together with the very low barriers to entry created by the platform providers, has led to a rapid and very substantial expansion in the number of overall apps."[244] Google's Mr. Rutledge testified that in the 2008 to 2009 time frame, Google "had quite a hard time getting any interest of any application developer to go for Android…Primarily [because of] the size of the device ecosystem. We had no market share."[245]

164. Android was initially launched in October 2008 on a single HTC handset and at a single carrier in the U.S., T-Mobile, which at the time was only the fourth largest U.S. carrier with a 12% share of the mobile phone market.[246] Given the limited distribution potential, the Android installed base was still small one year following the launch of the first Android handset.[247] Also contributing to the low level of installed base were limitations of the HTC handset and Android 1.0 itself.[248]

---

[243] Meier 12/11/2015 Dep., pp. 43:10-46:24. Mr. Meier noted that large companies, in particular, could afford to wait until Android had built a user base before investing in an Android app (pp. 52:19-53:24). Yet, such companies are those that provide some of the most popular apps as discussed elsewhere in this report.

[244] Bresnahan, et al. (2015), at location 5883 (Kindle version).

[245] Rutledge 12/9/2015 Dep., pp. 77:17-23. See also Rutledge Dep. Ex. 5057 ("Developers need to see volume shipments before they write apps."); Rutledge 12/9/2015 Dep., p. 109:03-110:10; Meier 12/11/2015 Dep., pp. 40:8-41:21; "Additional Item: If Android Does Succeed It Will Be By Quite a Narrow Margin," Optical Networks Daily, August 7, 2009.

[246] "A Brief History of Android Phones," CNET, August 2, 2011, http://www.cnet.com/news/a-brief-history-of-android-phones/; Fourteenth Annual Report and Analysis of Competitive Market Conditions With Respect to Mobile Wireless, Including Commercial Mobile Services, Federal Communications Commission, May 20, 2010, p. 31.

[247] Exhibit 2h.

[248] Regarding limitations of the handset, see "T-Mobile G1 review," CNET, October 15, 2008, http://www.cnet.com/products/t-mobile-g1/ ("for now, the [HTC handset] is best suited for early adopters and gadget hounds, rather than consumers and business users"). When Android 1.5 was released, it was viewed as a major improvement over 1.0, demonstrating the limitations of 1.0. See "Top 10 Features You'll Love about Android 1.5," http://www.geek.com/android/top-10-features-youll-love-about-android-15-768061/ (1.5 "significantly enhance[s] the [user] experience, addressing most glaring omissions from the 1.0 release");

165. The limited initial availability of Android devices, and the resulting slow initial growth of the Android user base, also are inconsistent with the suggestion by Oracle's experts Dr. Kemerer and Mr. Malackowski that Google faced a limited "window" of opportunity to launch Android, and that if Google missed that window, Android would have failed. Oracle's experts use this suggestion to argue, in essence, that had Google not used the allegedly infringing works in this case, Android would not exist today, or in any event that Google would have earned substantially less in profit from Android than it has. However, I note that none of Oracle's experts provide any economic analysis establishing that such a "window" existed, much less showing when it began, when it ended, and its significance. Oracle's experts rely on contemporaneous Google documents discussing Google's perceptions of the market opportunity, but do not appear to have done any analysis to verify whether and to what extent Google's perception was accurate or that the window would have soon closed. In fact, to the extent there was any such window, it was reasonably wide in terms of time, as demonstrated by the fact that, even a year after the Android launch, the Android user base was still quite small, owing in part to the limited number of Android devices in the market with smaller carriers. Yet, the Android user base grew substantially in subsequent years, particularly after the introduction of additional, and higher-end, Android devices with the larger carriers

---

"Google Phone Update: Android 1.5 'Cupcake' Reviewed," http://www.informationweek.com/mobile/mobile-devices/google-phone-update-android-15-cupcake-reviewed/d/d-id/1080104?page_number=1 ("In sum, the 1.5 update to Android makes the platform much stronger. It adds a host of missing features, makes improvements to the overall usability of the operating system, and makes the platform feel more complete"); "Google Android 1.5," http://www.pcmag.com/article2/0,2817,2349058,00.asp ("The latest version of the OS, Android 1.5, features a number of major enhancements").

beginning in late 2009, two years after the introduction of the Android platform. Thus, the "window" had not closed within at least two years after Android's actual introduction.

166. Just prior to Android's launch, Google removed some apps from the Android app store, which reduced the number of apps in the store to 13.[249] The number of new apps added to the store per month remained at low levels during the first year after the Android launch.[250]

167. In October 2009, the distribution potential for Android increased considerably when Samsung and Motorola introduced their first Android handsets in the U.S. and these handsets were available with the carriers Sprint and Verizon, which were the third and first largest carriers in the U.S.[251] Moreover, these handsets were well-received and the Motorola Droid had the newest version of Android, Android 2.0 (the Samsung handset had Android 1.5, which was still a major improvement to Android 1.0).[252] The Android installed base picked up after this point in time, and this in turn was followed by a jump in the number of new apps per month.[253]

---

[249] "Google Removes Applications Just Before Launch," Android Community, http://androidcommunity.com/google-removes-applications-just-before-launch-20081020/. Shortly after the Android launch, reports suggest that there were about 62 apps in the Android app store. "Top Ten Android Launch Apps," Tech Crunch, http://techcrunch.com/2008/10/22/top-ten-android-launch-apps/.

[250] Exhibit 2g.

[251] "A Brief History of Android Phones," CNET, August 2, 2011; Sixteenth Annual Report and Analysis of Competitive Market Conditions With Respect to Mobile Wireless, Including Commercial Mobile Services, Federal Communications Commission, March 21, 2013, p. 54.

[252] "Motorola Droid (Verizon Wireless) review," CNET, October 28, 2009, http://www.cnet.com/products/motorola-droid-verizon-wireless/; "First Google Android 2.0 Phone Arrives," CNET, October 30, 2009, http://www.cnet.com/news/first-google-android-2-0-phone-arrives/; "Top 10 Features You'll Love about Android 1.5," Geek, May 8, 2009, http://www.geek.com/android/top-10-features-youll-love-about-android-15-768061/; "First Look – Samsung I7500 Preview," GSMArena, May 21, 2009, http://www.gsmarena.com/samsung_i7500-review-351.php; "Google Phone Update: Android 1.5 'Cupcake' Reviewed", InformationWeek, June 2, 2009, http://www.informationweek.com/mobile/mobile-devices/google-phone-update-android-15-cupcake-reviewed/d/d-id/1080104?page_number=1.

[253] Exhibits 2g, 2h.

168. This is an example of how, as discussed in greater detail below, the growth in Android users was driven to a large degree by improvements in the price-adjusted quality of the devices running Android and additional functionalities provided by Android itself rather than the available apps. These hardware and OS improvements reflect the contributions of the OEMs and Google as well as the open source strategy of Google that resulted in a low price for the OS and thus encouraged OEMs to adopt Android.

e. Economics Literature on Platform Choice by App Developers and Smartphone Choice by Users

169. Several recent studies have examined platform choice by mobile app developers. A 2015 study by T. Bresnahan, et al. examines issues related to platform choices of app developers. Their empirical results are "consistent with the theory that the most important determinant of developer behavior is the installed base."[254] They find that many developers write apps for both Android and iOS, which is consistent with the conclusion that "the technical entry barriers to either platform are low for developers."[255] The technical entry barriers would include the software development and thus, in principle, could be affected by the choice of programming language. However, the fact that the technical entry barriers are low suggests that any such affect is minimal. The study itemizes a number of "asymmetries" between the Android and iOS platforms, one of which is that Java is "popular with developers"; however, the authors focus on Android's openness as the primary important advantage over iOS.[256] Finally,

[254] Bresnahan, et al. (2015), at location 6527 (Kindle version).

[255] Id. at location 6527 (Kindle version). The authors conclude that the marketing costs, however, are an important entry to barrier for apps. For example, getting on top app lists is both important for generating demand and costly. Such marketing costs are not related to the specific applications programming language.

[256] Id. at locations 6527, 6557 (Kindle version).

the study finds a high prevalence of firms with an app on Android to also offer an app (perhaps not the same app) on iOS; the authors interpret this result as evidence against the hypothesis that firms that develop for Android do not have the capabilities to develop for iOS (and vice-versa).[257] Instead, many firms have the capability to develop apps for both platforms. In that environment, Android's use of Java provided little advantage. Because it is "the most important apps" that are developed for both platforms[258], this suggests that the developers of the most important apps were not encouraged to develop for Android by Android's use of Java as an applications programming language.

170. A 2014 study by T. Bresnahan, et al. models a developer's platform choice. They note that the incentives to develop an app for a platform depend on the size of the user base, the expected penetration rate, the profits per user, and the fixed costs of development. They note that the proposition that fixed costs of development are approximately equal across Android and iOS is "completely consistent with what industry participants tells us."[259] It is not consistent with Java being responsible for a substantially lower cost of development on Android.

171. A 2014 study by Y. Liu examines the platform choices of game developers.[260] The developer choice is modeled on the profits that the developer expects to earn on each platform, which in turn depends on the size of the user base, the quality of the developer's app, the quality and number of competing apps on the platform, and the costs of developing the

[257] Id. at location 6695 (Kindle version).

[258] Bresnahan, et al. (2014), p. 1.

[259] Id. at fn. 13.

[260] Y. Liu, "Mobile App Platform Choice: An Application of Strategic Games on Big Data," November 2014.

app. Consistent with other studies, Liu finds that a small percentage of game apps are of high quality, with most of low to moderate quality[261], and that the size of the user base is a powerful driver of developer platform choices.[262]

172. A recent economics study estimated an econometric model of demand for smartphone operating systems.[263] Demand for a smartphone OS was specified to depend on the characteristics of mobile devices (such as screen size and price) that ran the OS[264] as well as the expected utility for a user from the applications available on the OS. The model was estimated on data covering the period January 2010 to December 2012, a period of time in which Android experienced a substantial growth in users. The results of the study found that differences in the available applications explained little of the relative changes in user shares between Android smartphones and iPhones, despite large changes in the number of available Android applications over the period; instead, Android's stronger user share growth was due to be driven primarily by the relative improvements in the price-adjusted quality of mobile devices running Android. This result is consistent with users driving app development, rather than the reverse, and the user value from apps being concentrated in a small number of apps (in which case a growing number of apps would not affect user demand for Android significantly).

[261] Id. at p. 22 ("the app market is highly skewed toward a few killer apps…").

[262] Id. at p. 30 ("before Google started to take off in 2011, small Android phone user base as associated with the slow growth of [the] Google app store").

[263] M.J. Kim, Essays on the Economics of the Smartphone and Application Industry, University of Minnesota, Ph.D. Thesis (2013), hereafter "Kim (2013)."

[264] Other studies have found that device characteristics such as display, camera, sensors, and brand name influence a user's choice of device. See Chow, et al., Factors Affecting the Demand of Smartphone Among Young Adult, Multimedia University (2012).

4. Conclusion Regarding the Contribution of the Alleged Infringement

173. Based on the foregoing, I conclude that the contribution of the alleged infringement has been small relative to the contributions of other factors that were substantially more important to Android's success, and thus to causing the revenues Mr. Malackowski claims are attributable to the infringement. This implies that the apportionment of the Android-related profits to the alleged infringement should be small as a matter of economics.

E. Quantitative Approaches to Apportionment

1. Bottom Up Approaches: Direct Measurement of the Android-Related Profits Attributable to the Alleged Infringement

174. A "bottom up" apportionment works by directly measuring the portion of the Android-related profits that are attributable to the alleged infringement. As discussed above, the contribution of the alleged infringement to Google's Android-related profits was that it generated cost-savings for Google by allowing Google to avoid taking certain costly actions such as licensing the allegedly infringing work under the OpenJDK and implementing the 37 API packages prior to the Android launch, paying for developers to be trained in another programming language, or paying for application development. I discuss several approaches to measuring these cost-savings. I also analyze the contribution of the alleged infringement to the Android-related profits under the assumption that Google took none of these costly actions. I find this measure to be small relative to the Android-related profits, but still larger than the cost-savings associated with the avoided actions described above. Thus, the appropriate apportionment should be based on the cost-savings and, in fact, the smallest of the cost-savings.

a. Avoidance of Costs Associated With Switching to the OpenJDK

175. The alleged infringement allowed Google to avoid costs associated with licensing the allegedly infringing work through OpenJDK, and then implementing that licensed solution, prior to Android's launch.

176. Android was announced on November 5, 2007; the Android SDK was released shortly thereafter; and the first Android handset was introduced on October 22, 2008. Sun released the OpenJDK on May 8, 2007, approximately six months prior to the Android announcement. The OpenJDK still had certain encumbrances at that point—specifically, a relatively small amount of the code was still proprietary[265]—but these encumbrances did not involve the SSO and declaring code for the 37 API packages at issue.[266] Thus, Google had approximately six months to implement the OpenJDK class libraries for the 37 API packages at issue in Android before the Android announcement and release of the Android SDK. I understand that Google has recently actually implemented the OpenJDK class libraries for the 37 API packages at issue and that this implementation was accomplished in approximately a six

[265] "Sun Announces Open Source Community Innovation Awards Program; Multi-Year Program Expected to Payout Millions to Foster Global Community-Based Innovation," Business Wire, December 5, 2007; "Sun Fulfills Promise of Open and Free Java Technology and Releases Java SE Platform to OpenJDK Community; Advances OpenJDK Project with New Code, NetBeans Integration, Governance Board and Availability of Compatibility Tests;" PR Newswire; May 8, 2007; "IcedTea: The First 100% Compliant Open-Source Java," InfoQ News, June 21, 2008, available at http://www.infoq.com/news/2008/06/icedtea_tck; "Open-source Java could result in port to iPhone; As the first anniversary of open-source Java approaches and Apple prepares to release an SDK for the iPhone, Sun thinks it may have a way to put Java on the popular handheld," InfoWorld Daily News, November 9, 2007.

[266] OpenJDK-based implementations free of any encumbrances were available by April 30, 2008. "Open Source Java Technology Debuts in GNU/Linux Distributions; Latest Releases of Fedora and Ubuntu Feature OpenJDK-Based Implementations," Business Wire, April 30, 2008.

month period with, for the most part, a single engineer assigned to the project.[267] I further understand that the time frame could have been compressed to no more than three months by adding another engineer to the project.[268] Moreover, to implement the OpenJDK class libraries for the 37 API packages at issue prior to Android's launch would have been substantially easier than it was now because of the reduced complexity of Android at the earlier point in time.[269] Therefore, I conclude that Google could have implemented the OpenJDK class libraries for the 37 API packages at issue starting in May 2007 and not caused any delay in the November 2007 Android announcement and release of the SDK, let alone the October 2008 Android launch a year later.[270]

177. Moreover, I understand that Google could have taken and used just the SSO and declaring code for the 37 API packages from OpenJDK and combined this declaring code with Google's own implementation code.[271]

178. The incremental cost to Google of implementing the OpenJDK libraries would have been about one engineer's time for six months (or two engineers' time for three months). The average salary of a senior software engineer in Silicon Valley in 2008 was $115,730 and,

---

[267] Interview of Anwar Ghuloum. I understand that Mr. Ghuloum was Google's Rule 30(b)(6) designee witness on issues related to Google's implementation of the OpenJDK solution and was prepared to testify at deposition to the facts conveyed to me regarding the time and effort expended by Google in implementing that solution. However, counsel for Oracle did not ask questions on that subject when Oracle deposed Mr. Ghuloum.

[268] Interview of Anwar Ghuloum.

[269] Interview of Anwar Ghuloum.

[270] Oracle's expert Dr. Kemerer claims that the GPL with Classpath exception license would have been unacceptable to Google. He is obviously incorrect given that Google has in fact switched to the OpenJDK implementations. He is also incorrect because he seems to confuse the GPL license with the GPL with Classpath exception license. Parts of Android are licensed under the latter, so again actual events demonstrate that this license would be acceptable to Google. See Ghoulom 12/9/2015, Dep., pp. 33:6-35:25.

[271] Astrachan 1/8/2016 Report, ¶ 258.

assuming a benefits to salary percentage of 31.7%, the incremental cost would be $84,722. The total cost savings, and thus unjust enrichment (assuming infringement), that Google received from using the allegedly infringing material is approximately $85,000.[272]

b. Avoidance of Costs Associated With Developer Training

179. Another measure of the contribution of the alleged infringement is the cost that Google would otherwise have had to incur to train third party app developers who were unfamiliar with an alternative programming language, such as C or C++. Note that no developer training would be needed if Google had implemented the OpenJDK class libraries. Thus, developer training cost-savings is an alternative measure of the unjust enrichment (assuming infringement).

180. Based on the App Annie data, I have determined that during the period 2012-2015, there were 1,889 developers that had an Android app in the daily top 100 download list for at least one day.[273] Given the extreme skew in the demand for apps (i.e., a few apps are highly demanded and most are marginal; 200 apps account for the large majority of usage, etc.), this list captures the developers of important Android apps. Some of these developers used C/C++ to develop their apps and thus would not need to be trained in C/C++. I have used a list of the Java/NDK status of apps provided by Google to eliminate developers that knew C/C++ already (this is a conservative approach given that this Java/NDK status list did not include all apps; if an app is not on the list, I assume it is Java). I also eliminated developers that

[272] Exhibit 3a.

[273] The market intelligence data from App Annie provides daily ranking, number of downloads, and developer information for the top 100 free and paid apps in iOS and Google Play between January 2012 and December 2015. The data also identifies the different versions of an app across and within platforms. "App Annie Intelligence Product Suite Overview," App Annie.

multi-homed on iOS and thus had demonstrated an ability to develop in multiple languages. After those two sets of developers are eliminated, 986 developers are left. Some of these developers may well have already known C/C++ or been willing to learn another language without being paid by Google. However, to derive a conservative estimate of the savings in training costs, I assume all of these developers would need to be trained. I double this number to account for the entire 2008-2015 time period. Finally, assuming 1.6 programmers per developer on average[274], there would be a maximum of 3,155 programmers that would need to be trained in C/C++. There are many C/C++ courses available for free; as discussed above, paid courses cost approximately $715.[275]

181. Thus, a conservative measure of the cost-savings for Google from having to avoid training developers in another programming language such as C/C++ is $2,256,000.[276]

c. Avoidance of Costs Associated With Paying Third Party Developers to Develop Android Apps

182. Another measure of the contribution of the alleged infringement is the cost that Google avoided having to incur to promote app development by third party developers. Note that paying for app development would not be needed if either Google had implemented the OpenJDK class libraries or Google had provided training to developers unfamiliar with C/C++.

[274] There were a total of 73 developers working on the 46 applications made public in the list of Top 50 Android Applications for Google's Developer Challenge in 2008. This gives an average of 1.6 developers per app. See "Google Reveal The Top 50 Android Applications (46 Public)," Chris Moor at Talk Android, http://www.talkandroid.com/92-developer-challenge-top-50-android-application/; "Splah Play," AndroidTapp, http://www.androidtapp.com/splashplay/.

[275] Exhibit 3c.

[276] See Exhibit 3c.

Thus, the cost-savings related to paying for app development is an alternative measure of the unjust enrichment (assuming infringement).

183. As discussed above, estimates of the costs of app development range from $25,000 to $100,000. As also discussed above, a relatively small number of apps account for the large majority of usage. Some of these apps would be written in C/C++ in any event (so that Google would not need to pay for development), and others would be written be developers (e.g., Facebook) who would not need any additional incentive to develop Android apps even in a different language. I estimate that, over the time period in question, there would be no more than 1000 "important" non-C/C++ apps.[277]

184. Assuming, quite conservatively, that Google would have had to pay for development of all these apps, the cost-savings to Google from the alleged infringement (and thus the unjust enrichment) would be between $23 million and $100 million, depending on the cost of app development.[278]

d. Incremental Android-Related Profit Attributable to the Alleged Infringement, Assuming No Costly Google Actions

185. As discussed above, the alleged infringement generated cost-savings for Google by allowing Google to avoid taking certain costly actions. If Google did not take such actions and did not allegedly infringe, Oracle's claim is that there would have been fewer Android apps

[277] Based on the comScore data, there are 428 unique apps that were among the monthly top 200 most used apps during the January to March 2013 and January to March 2015 periods combined. Using an analysis of how long apps tend to remain in the top 200, I estimate that no more than 1,900 apps would be among the monthly top 200 most used apps over the course of the Android's existence. I conservatively estimate that 65% of these apps would be Google apps, C++ apps, or apps by multi-homing developers. Thus, Google would have to cover the development cost for a maximum of 665 apps. I further conservatively round this up to 1000 apps.

[278] See Exhibit 3b.

and this would have impacted Android device sales and therefore the Android-related profits. In this section, I measure these effects.

186. I have applied the Kim (2013) empirical model of smartphone demand conservatively to estimate the decrease in Android handset sales that would have occurred in a counterfactual where there were fewer Android apps, as well as the percentage of this Android sales decrease that would have been captured by the iPhone. Google would earn ad revenue on these additional iPhone units.

187. In the model, the Android share satisfies (after application of the Berry (1994) inversion)[279]

$$\ln(s_A) - \ln(s_0) = \delta_A + \sigma \ln(s_{A|A,I})$$

where s_A is the Android share, s_0 is the share of the outside good, δ_A is the mean utility level for Android (based on the right-hand-side of equation (2.3) of Kim (2013)), and $s_{A|A,I}$ is Android's share of a nest consisting of Android and iPhone. iPhone share is defined analogously as

$$\ln(s_I) - \ln(s_0) = \delta_I + \sigma \ln(s_{I|A,I})$$

Blackberry share is defined as

$$\ln(s_B) - \ln(s_0) = \delta_B$$

and the share of the outside good is defined as $s_0 = 1 - s_A - s_I - s_B$.

188. Given data on shares and the value of σ[280], the three mean utility levels δ_A, δ_I, and δ_B can be solved for.

[279] S. Berry, Estimating Discrete-Choice Models of Product Differentiation, Econometrica (1994). In these equations, I omit notation related to time for clarity. The analysis is conducted on a monthly basis.

189. In the counterfactual where some Android apps are not available, the Android app value variable in the Kim (2013) model (I will call it V_A) takes on a different value from its actual value. Given a set of available Android applications, V_A is defined as

$$V_A = \sum_{k=1}^{K_A} \ln(1 + \exp(\theta_k))$$

where K_A is the number of available Android apps and θ_k is the mean utility level for app k (based on equation (2.7) of Kim (2013)). V_A is well approximated as[281]

$$V_A = \sum_{k=1}^{K_A} \ln(1 + \frac{s_k}{1 - s_k})$$

The decrease in V_A (i.e., the decrease in the Android app value variable) given a smaller set of available Android applications is

$$\Delta_A = \sum_{k \in M} \ln(1 + \frac{s_k}{1 - s_k})$$

where M represents the subset of apps assumed not to be available in the counterfactual. Then, the counterfactual value of the Android mean utility level δ_A' can be calculated as

$$\delta_A' = \delta_A + \beta \Delta_A$$

where β is the coefficient on V_A.[282] The iPhone and Blackberry mean utility levels do not change in the counterfactual.[283]

---

[280] Kim (2013) estimated this parameter to be 0.757.

[281] In a binary logit model, the aggregate share $s = \exp(\theta)/(1 + \exp(\theta))$ where θ includes an unobserved app quality effect, as in the app demand model in Kim (2013). Then, by the Berry (1994) inversion, $\exp(\theta) = s/(1 - s)$.

[282] Kim (2013) estimates this parameter to be 0.01.

[283] Note that, in the context of this model, determining the effect of a change in a single explanatory variable (in this case the Android app value variable) requires knowledge only of the actual value of that variable, its coefficient, the nesting coefficient, and the actual shares of each of the operating systems. It is not necessary

190. The counterfactual shares can be calculated by solving the following system of equations.

$$\ln(s'_A) - \ln(s'_0) = \delta'_A + \sigma \ln(s'_{A|A,I})$$

$$\ln(s'_I) - \ln(s'_0) = \delta_I + \sigma \ln(s'_{I|A,I})$$

$$\ln(s'_B) - \ln(s'_0) = \delta_B$$

$$s'_0 = 1 - s'_A - s'_I - s'_B$$

191. In the Kim (2013) model, the variables are defined as follows.

- The share of a smartphone OS in a given month is defined as the U.S. unit handset sales in that month divided by the U.S. population over the age of ten. I used U.S. unit handset sales data from ITG and U.S. population data from the U.S. Census.

- An app's share is defined as the number of downloads of the app in a given month divided by the handset sales of the OS in that month. I obtained app download data from AppAnnie. These data cover the period January 2012 to December 2015. Kim (2013) identified the set of available apps for Android and iPhone based on top paid and free app lists. I used the same approach.

192. For each Android app, I determined whether (1) it is a Google app, (2) it was written using the NDK, (3) it was multi-homed on iOS, (4) its developer also developed apps for iOS, or (5) its developer also developed NDK Android apps. Any app in one or more than one of these categories is assumed to be available on Android in the counterfactual. Apps in none of these categories are assumed to be unavailable in the counterfactual. This is conservative because apps not in one of these categories may well have been developed anyway. For example, many apps not in these categories have close counterparts on Microsoft Phone, which

---

to know the values of the other individual explanatory variables; their combined value is all that is needed and this can be determined through the Berry (1994) share inversion.

has much smaller user base than Android.[284] Based on this assumption, the counterfactual shares are calculated as described above.

193. Given the counterfactual shares for each month, I calculate (1) the decrease in Android's sales and (2) the increase in iPhone sales in the counterfactual relative to their actual levels.[285] I then take appropriately weighted averages to calculate by year (1) the percentage Android sales decrease in the counterfactual and (2) the percentage of the Android sales decrease in the counterfactual that is captured by the iPhone (the "diversion ratio").[286]

194. The calculations described above are based on U.S. shares. I repeated the exercise after adjusting the shares to reflect rest of world (ROW) shares. Because the iPhone's ROW share is lower than its U.S. share, the ROW diversion ratio is lower and the Android unit loss percentage is lower.[287] The U.S. and ROW diversion ratios and Android unit loss percentages are averaged using Google U.S. and ROW Android-related ad revenues as weights, yielding a percentage loss and diversion ratio that can be applied to Google Android-related ad revenue. I similarly calculate a percentage loss that can be applied to Google Play revenue.

195. I then apply the percentage loss in revenue and diversion ratio figures and calculate the appropriate costs to determine the apportionment of the Android-related profits

---

[284] Exhibit 2i.

[285] Exhibit 3d.3. In principle, there could be "feedback" from the decrease in users to apps, which could then lead to a further decrease in users. However, this feedback effect is negligible for several reasons. First, the initial decrease in users is small, so any feedback effect would likewise be limited in size. Second, given that developers focus on return on investment, the initial small decrease in users would affect only those apps that are marginal, i.e., those just on the boundary of being profitable. But, these apps by definition are those with a very low user share and thus their absence would have little additional effect on the number of users.

[286] Apple would have had time to expand its supply of iPhones and, indeed, the capacity in the various component industries made available by the lower Android handset sales could have been utilized by Apple.

[287] Exhibit 3d.3.

to the alleged infringement, assuming Google does not take any of the costly actions described in the previous sections.[288] I find this figure to be no more than $203 million. For the reasons described above, it is highly conservative.

196. The appropriate measure of the apportionment of Google's Android-related profits to the alleged infringement using the bottom-up approach is the minimum among the three cost-savings and the profit loss.

2. Top Down Approaches: Application of an Apportionment Percentage to the Android-Related Profits

197. In contrast to the "bottom up" approaches to apportionment discussed above, a "top down" approach to apportionment involves identifying the percentage of the Android-related profits that should be attributed to the alleged infringement as opposed to the contributions of other parties and factors, and then applying this percentage to the Android-related profits.

198. I have taken two approaches to top down apportionment. Under the first approach, in a first step I apportion between Android, on the one hand, and Google's search and ad technologies and services, on the other, as described above. This yields a [redacted] apportionment of profit to Android. In a second step, I apportion between the 37 API packages, on the one hand, and the remainder of Android, on the other. I have seen no evidence that the programming that went into the allegedly copyrighted material reflects any higher level of ingenuity than the programming that went into the other parts of Android.[289] The 37 API

[288] Exhibit 3d.1.

[289] Astrachan 1/8/2016 Report, ¶¶ 119-120, 141.

packages at issue are "fundamental" to the Java language, which means they carry out fundamental and, at some level, "mundane" tasks that programmers often need to perform, and also suggests that, without using the 37 API packages, it would be difficult for a programmer to use the Java language effectively at all.[290] However, the other API packages that Google developed for Android similarly can be described as fundamental to the application programmer's experience, if not as essential to use of the Java language. API libraries are often created for programming languages (and platforms) as a convenience to programmers so that each programmer need not write their own version of code to run these fundamental tasks. The creation of such libraries of APIs is in no way unique to the Java programming language or to the SSO and declaring code of the 37 API packages asserted by Oracle in this case. Indeed, the libraries available for other programming languages provide similar functionalities and in fact the allegedly copyrighted declaring code for the 37 API packages in some cases borrows from other languages that pre-date Java.[291] Thus, at a high level, there is nothing significant that is unique about the 37 API packages at issue from a programming language perspective.

199. Given the lack of evidence that the 37 API packages at issue are unique or reflect a special level of programming ingenuity[292], a reasonable approach to the apportionment between the 37 API packages and the rest of Android can be based on programming effort, which in turn can be measured by lines of code or number of methods coded.

[290] Astrachan 1/8/2016 Report, ¶ 17. On the close connection between the language and the APIs, see Smith 11/20/2015 Dep., pp. 22:16-24:8, 288:14-292:19.

[291] Astrachan 1/8/2016 Report, ¶¶ 18, 203-220.

[292] Astrachan 1/8/2016 Report, ¶¶ 154, 182-202.

200. According to an Oracle expert witness, the 37 API packages at issue include 12,774 lines of declaring code allegedly subject to copyright.[293] Dr. Astrachan found that the 37 API packages contain 259,474 lines of total code, including both the relatively small number of lines of code in the accused SSO and declaring code and the much larger number of lines of non-accused, Google-written implementing code.[294] Android overall has 15,347,169 lines of code.[295] Using these figures, I calculate an apportionment percentage for the alleged infringement to be no more than 1.7%, based on number of lines of total code in the 37 API packages at issue versus all of Android.[296] This is conservative because it includes the Google-written implementing code in the numerator, whereas I understand that only the SSO, as expressed through the declaring code, is accused of infringement in this case. If only the SSO and declaring code were used in the numerator, the percentage would be only 0.08%.

201. As discussed above, the 1.7% apportionment percentage should be applied to the [redacted] that was apportioned to Android in the first step. This results in a measure of unjust enrichment (assuming infringement) of [redacted].[297]

202. Under the second top-down approach, I calculate an apportionment percentage where the numerator is the lines of declaring and implementing code in the 37 API packages and the numerator is the sum of the lines of code in Android overall, Google's primary search code base (1.7 million lines), and Google's primary ads code bases (48.5 million lines). The

[293] Zeidman 1/8/2016 Report, ¶¶ 43, 45.

[294] Astrachan 1/8/2016 Report, ¶ 140.

[295] Astrachan 1/8/2016 Report, ¶ 140.

[296] Exhibit 3e.

[297] Id.

latter two are conservative estimates as they only account for Google's primary search and ads code bases, but do not include the large additional code bases that support Google's ads and search technologies. This yields an apportionment percentage of 0.4%, which when applied to the [redacted] Android-related profit figure, gives a measure of unjust enrichment (assuming infringement) of [redacted][298]

F. Conclusion on Unjust Enrichment

203. The economic evidence strongly supports the conclusion that the apportionment to the alleged infringement should be small. The Android-related revenues and profits are the result of contributions from many factors other than the alleged infringement. In addition, the contribution of the alleged infringement was minor. Most apps are not highly used. Most of the relatively small number of apps that are highly used were developed by Google itself or by companies such as Facebook that have both a strong economic incentive to write mobile apps and a demonstrated ability to write apps in other languages. While Java may have been more familiar to some programmers, there were many programmers who were familiar with C, C++, and other languages. Multi-homing by developers is common. Moreover, programmers can easily learn new languages if the economic incentive to do so exists. Finally, the value of the contribution of the alleged infringement to Google was that Google could avoid incurring expenses to implement the OpenJDK prior to the Android launch, train developers, or pay for app development.

204. Based on bottom up approaches to measuring the apportionment of the Android-related profit to the alleged infringement, I conclude that the apportionment to the

[298] Id.

alleged infringement should be no more than $100 million and should in fact be much lower. Based on the top down approaches, I conclude that the apportionment to the alleged infringement should be no more than $56 million.

G. It Would Be Economically Inappropriate to Base An Apportionment Percentage on Google's Split of Ad Revenue With Apple (or Handset OEMs or Carriers) For Ads on Apple Devices

205. I understand that Oracle has sought information on how Google splits ad revenues with partners such as handset OEMs and carriers. In doing so, Oracle has suggested that Google's partner agreements somehow suggests an amount, or a magnitude of value, that Google was willing to pay to make sure that Google's search engine was available to users of other OEMs' devices.

206. It would be economically inappropriate to apportion to the allegedly infringing material the same percentage of ad revenues that, say, Apple receives for ads on the iPhone. To do so would be tantamount to assuming that the contribution of the allegedly infringing material is equivalent to the entirety of Apple's contributions. This is unreasonable from an economics point of view. The SSO and declaring code of the 37 API packages at issue are not in any way equivalent to Apple's contributions, which include, among other things, developing the iOS operating system, designing and supplying the iPhone device hardware, providing an applications development framework and an app store, developing numerous applications, and using the well-known Apple brand name to promote its products. Instead, a more appropriate comparable for the allegedly infringing material is a subset of APIs within the iOS application development framework. From that perspective, it can be seen that the appropriate

apportionment percentage for the allegedly infringing material is only a small percentage of the total Apple apportionment percentage.

VII. Oracle's Claimed Actual Damages: Lost Profits

A. Overview

207. Oracle's actual damages can be calculated based on Oracle's actual lost profits, if any, that were caused by Google's alleged infringement.

208. Oracle's damages expert, Mr. Malackowski, calculates Oracle's actual damages by applying "a 'but-for' test which quantifies a portion of the additional profit Oracle would have achieved absent Google's infringement (i.e., Oracle's lost profits). [His] quantification of Oracle's lost profits is based on a comparison of the actual results generated by Oracle's licensing of Java ME, relative to the results it would have been generated 'but-for' Google's infringement."[299] Mr. Malackowski inappropriately attributes the entire difference between Oracle's "but-for" profits from licensing Java ME and Oracle's actual profits from licensing Java ME to Google's alleged infringement. I note that Mr. Malackowski's calculation of lost profits damages is based on lost profits of Java ME, which is not even the copyrighted work at issue in this matter. In fact, an app written in Java ME, a simpler platform with relatively few APIs designed for use on smaller, less sophisticated mobile devices, would not even work on an Android device or any other smartphone product. Indeed, an Oracle witness stated that Java ME "was not fully capable for what's required by a smartphone device."[300] Thus, there is a disconnect between Java ME—the product that Mr. Malackowski is claiming lost sales, and thus

[299] Malackowski 1/8/2016 Report, ¶ 180.

[300] Rizvi 7/28/2011 Dep., p. 203:5-7.

profits, as a result of Google's conduct, and the accused product in this matter—Android. Put another way, Oracle is claiming that Android caused it to lose sales of a product, Java ME, that does not compete with Android because it has proven to be unsuitable for the devices (smartphones and tablets) for which Android was designed.

209. Mr. Malackowski's calculation of lost profits damages suffers from another fundamental error. As will be discussed below, his calculation of what would have happened in the "but-for" world—a world where the only change from the underlying economic environment of the actual world is the fact of Google's alleged infringement—is based on unreasonable Java ME licensing forecasts that Sun would never have been able to achieve due to several reasons not factored into these forecasts including, for example, Sun's open sourcing of Java ME, the stagnation of Java ME as a result of Sun's lack of investment, the unexpectedly large shift towards smartphones, and the 2008 financial crisis. Mr. Malackowski's approach to calculating lost profits damages, which attributes the entire difference between (1) these forecasts that failed to anticipate actual world events that would have adversely affected Sun's Java ME licensing revenues and (2) Sun's actual Java ME licensing revenues to the alleged infringement, therefore, significantly overstates damages. The proper way to calculate lost profits damages is to look at what actually happened, not what was forecasted to happen, and to isolate the effect, if any, that the alleged infringement actually had on Oracle's Java ME licensing revenues and profits, taking account of economic factors other than the alleged infringement that affected those revenues and profits.

210. Additionally, Mr. Malackowski "considered the negative impact of Android on Oracle's ability to launch project Acadia,"[301] but does not quantify this "negative impact."

211. As I will discuss throughout the following sections, Sun's lost Java ME licensing revenues and profits, and the failure of Sun's Java FX / Project Acadia, were a result of factors having nothing to do with Google's alleged infringement or, for that matter, Google's introduction of Android. For example, first it is notable that there is substantially less than complete overlap between the Java ME APIs and the 37 Java SE APIs that are at issue in this case. Therefore, the Java ME APIs were not directly replaced by the 37 APIs at issue by any potential Java ME licensee. Second, and relatedly, given its limited set of APIs, Java ME is suitable for use only on feature phones, not smartphones, where Android is used. Given that smartphones substantially displaced the feature phones on which Java ME was used (and would have done so to largely the same degree even in the absence of the alleged infringement or Android), Sun's Java ME licensing revenues would have declined substantially over time in the absence of the alleged infringement. Third, in the but-for world, contrary to Mr. Malackowski's assumptions, Android still would have existed. Google could have chosen to use one of many other existing application programming languages and still achieved essentially the same level of success with Android as it did in the actual world using Java as the applications programming language. Google could have also used either the SSO and declaring code or even the complete implementations of the 37 API packages at issue from OpenJDK prior to the Android launch without causing any delay. Fourth, Sun and Oracle were both unable to introduce a

[301] Malackowski 1/8/2016 Report, ¶ 204.

smartphone operating system for reasons unrelated to Google's alleged infringement and Google's introduction of Android.

212. Given that the decline in Java ME licensing revenues, and hence profits, were caused by factors other than the alleged infringement and that any claim that Sun or Oracle would have successfully introduced a smartphone operating system is speculative, I conclude that Oracle's claimed actual lost profits damages due to Google's alleged infringement are zero. However, to be conservative, I have made adjustments to Mr. Malackowski's calculation of Oracle's claimed actual lost profits damages as further described in the sections below.

B. Sun Did Not Stand to Lose Substantially, and Even Could Have Gained, from Android

213. At the time that Google was developing Android, Sun was faced with a difficult competitive situation. Sun and Oracle witnesses stated that, at the time of first infringement and for many years thereafter, Sun's Java platforms, including the Java ME platform for the mobile space, were technologically "stagnant."[302] In addition, the Java ME platform was "fragmented" and was not well suited for, or widely used in, the then-emerging and now dominant market for smartphones. Given this situation, rather than viewing Android as an "incompatible" source of additional "fragmentation," Sun recognized that Android provided an opportunity to reduce the fragmentation that plagued Sun's Java ME in the mobile space. Perhaps even more importantly, Sun recognized the danger that its Java platforms were becoming outdated "legacy" technologies. By choosing to use the Java programming language, Android extended the relevance of the Java programming language in the mobile space.

[302] OAGOOGLE001208094.

Finally, other factors having nothing to do with Android ultimately contributed to Java ME's eventual decline, and Sun's and Oracle's inability to enter the smartphone market with a competitive mobile stack.

1. Sun's Java Platforms Were "Stagnant" Prior to the Introduction of Android

214. The Java programming language and Sun's Java platforms were "stagnant" even before the introduction of Android. Sun had failed to update and add to its Java platforms, and this led to dissatisfaction in the developer community and helped accelerate a shift in developer interest to other platforms.[303] The licensing revenue from the Java platforms was in decline as a result of competition from other platforms such as Linux.[304] Steve Jobs indicated that he was of the view that the Java programming language and Sun's Java platforms did not have a bright future. In 2007, Mr. Jobs was quoted as saying: "Java's not worth building in. Nobody uses Java anymore. It's this big heavyweight ball and chain."[305] "In 2010, when Jobs dropped Java like a hot cup of coffee, he tried to shame Oracle into supporting it. Since then, Java's been a neglected stepchild in the Mac world, completely shunned in iOS."[306]

---

[303] See, e.g., "What Are Your Java ME Pain Points, Really?" artima developer Weblogs Forum, http://www.artima.com/forums/flat.jsp?forum=106&thread=205707. A blogger posted on May 16, 2007 that "I did a lot of ME development about 4 or 5 years ago. I agree with all that has been written regarding pain points and barriers. What I find stunning is that it hasn't changed." Another blogger posted on the same date that "Java ME is stagnant and neglected."

[304] Schwartz 7/20/2011 Dep., pp. 29:24-31:15.

[305] "Apple;s Tim Cook wins where Steve Jobs failed: On Java," InfoWorld, April 30, 2012, http://www.infoworld.com/article/2617172/mac-os-x/apple-s-tim-cook-wins-where-steve-jobs-failed--on-java.html.

[306] Id.

215. Oracle also recognized the stagnation of the Java platform.[307] Sun's and Oracle's joint Form CO filing to the EC at the time of the acquisition discusses a "lack of leadership by Sun" in that "[m]any key participants in the Java Community Process ('JCP,' the Java governing organization), including Oracle, are frustrated by Sun's lack of momentum, which dissuades adoption of and contribution to Java standards."[308] However, the Oracle acquisition did not improve the situation. There was still a lack of direction in terms of future development in the Java platform and the developer community was still kept in the dark.[309]

216. The specific situation for Sun's Java ME platform—Sun's version of the Java platform for mobile and embedded applications—was even worse. According to Michael Ringhofer, VP of Worldwide Java Business at Oracle, in response to a question on whether he had heard from customers that Java ME was stagnant, he responded that customers were asking for updates to Java ME 6 due to the amount of time between Java ME 6 and Java ME 7 launches: "I heard that Java ME 6, you know, when are we getting updates, where is 7 – or when is the new version coming out. So, in general, to Java ME, I have never heard that. But on Java ME 6 to new version, there were comments."[310]

217. Furthermore, Java ME was widely recognized as being inadequate given the trends in the mobile phone industry. Former Sun/Oracle employee Craig Gering testified that one of the main weaknesses of the Java ME business model is that "we didn't have enough off-the-shelf implementations for all the different device types" in order to "get new customers to

[307] Rizvi 7/28/2011 Dep., pp. 105:7-106:7.

[308] OAGOOGLE0000140295 at 301.

[309] Kaul 8/5/2011 Dep., pp. 177:15-180:15.

[310] Ringhofer 12/2/2015 Dep., pp. 338:5-339:17.

adopt Java."[311] Indeed, Google is an example of a company that found the Java ME platform to be unsuitable to achieve its user experience goals. As a Sun employee, Eric Bergman, noted in a March 2007 email sent to other colleagues: "[w]hen Google tells us that it is too difficult or in some cases impossible to achieve their user experience goals using Java ME, and that they are looking at alternatives, it seems like a wake-up call. Much of what they told us we've heard before, perhaps some of it was new, but taken as a whole I don't think it can or should be ignored."[312] Thus, Sun itself was well aware that the Java ME platform was obsolete and that there was a diminishing interest in the Java ME platform in the developer community. Immediately after Google announced its launch of Android, one of Sun's own employees, Todd Fast, sent an email to other Sun colleagues, saying:

> The problem I see is that there is no mechanism to get community around Java ME—it's too fractious, arcane, hasn't given people the ability to do the things they want easily (3D, bluetooth, etc.), and it's been this way for years. People are pissed off, and I don't see that changing anytime soon. Furthermore, the carriers have simply made it too difficult to get ME apps onto your phone, and the hottest mobile device around—the iPhone—doesn't use Java, much less Java ME, and never will…. Java ME is last-generation technology.[313]

218. Terrence Barr, Senior Principle Technologies and Product Manager at Oracle, believed that Sun failed to evolve the Java platform quickly enough.[314] Furthermore, Sun's failure to innovate with respect to the Java platform quickly enough likely contributed to lost licensing opportunities, as Mr. Barr acknowledged.

---

[311] Gering 7/20/2011 Dep., pp. 88:10-89:24; see also OAGOOGLE0018885325 ("[Java platform for mobile] is old and behind in features").

[312] OAGOOGLE0000287870.

[313] OAGOOGLE0000293784.

[314] Barr 12/9/2015 Dep., pp. 157:7-164:17, 172:22-175:17, 246:10-248:11, 280:11-281:12.

> Q. And was Sun losing out on potential licensing opportunities by failing to innovate the Java platform quickly enough to address that market demand?
>
> THE WITNESS. In some instances, possibly.[315]

219. Additionally, Sun's business model no longer worked in the mobile space. As one Oracle document noted, "Update business models. OEMs are having difficult time to pay for platforms…Need solid partners…can['t drive a platform momentum/ecosystem alone…Market is trending toward verticalized solutions…We are providing a horizontal platform…"[316] Similarly, a November 2007 Sun email noted that "our mobile Java strategy is failing."[317]

220. According to Mr. Barr, Sun's failing Java mobile strategy had nothing to do with Android.

> Q. Prior to the release of Android on November 5th, 2007, did you believe that Sun's mobile Java strategy was failing?
>
> A. As I said before, I would qualify it and put it differently, but at the time, yes.
>
> Q. Okay. So that belief at that time had nothing to do with Android, correct?
>
> THE WITNESS. No. So I saw the market – and we spoke about that before – I saw the market being in a transition. I saw that the market was in need for more powerful and more capable mobile Java platforms.
>
> To me, Android was a – an instance of such a platform that required the mobile Java strategy at Sun to evolve. But it was not the only one.[318]

---

[315] Barr 12/9/2015 Dep., p. 174:15-20.

[316] OAGOOGLE0018885325.

[317] OAGOOGLE0008258138. On problems with Java ME, see also Barr 12/9/2015 Dep., pp. 158:19-159:15, 174:2-13, 180:3-19, 184:3-188:9, 198:10-199:8, 248:13-250:7, 288:16-290:2, 303:8-305:22; Barr Dep. Ex. 1369, 1371, 1372, 1373, 1378, 1382, 1385.

[318] Barr 12/9/2015 Dep., pp. 157:15-158:6.

2. Sun's Own Licensing Practices Caused the Java Platform to Be "Fragmented" Prior to the Introduction of Android

221. Sun's former CEO Mr. Schwartz testified that he did not view Android as fragmenting the "Java ecosystem."[319] Moreover, while some Sun employees may have expressed some general concern about the extent of the fragmentation of the Java ME platform, Sun's actual market actions speak louder than these words. Sun demonstrated that it was more than willing not only to accept, but to foster, "incompatibility" and the resulting "fragmentation" of Java ME in return for modest royalties.

222. For example, some of the early fragmentation of the Java ME platform can be attributed to Sun's inability to define completely the platform before the Java community began participating in and modifying the platform. According to Vineet Gupta in his deposition:

> Q. Did the fact that Sun was not able to completely define the ME platform before the community began participating in that platform lead to greater fragmentation in the ME platform than in the SE platform.
>
> A. I think it had certain amount of influence. But I think more influence was we assumed that companies would want to work together better, and what we found in this community is people were more willing to take a slower approach and willing to placate the ecosystem rather than advance the ecosystem.
>
> I think it also comes down to maturity. Desktops and Enterprise servers were already mature and consolidated. We consolidated the handset market, for example, and that has a bit to do with how platform evolution can happen or not.[320]

[319] Schwartz 7/20/2011 Dep., p. 215:14-17.

[320] Gupta 7/26/2011 Dep., pp. 267:15-268:5.

223. Indeed, Sun's own licensing policies had resulted in what was widely acknowledged to be a highly "fragmented" Java ME marketplace.[321] Sun licensed implementations that performed operations differently from each other.[322] Alan Brenner, Software Management Product Strategy Consultant for Oracle and former Vice President of Engineering for Java Card and Java ME, agreed that "Sun's liberal source distribution policy allowed development of fragmentation" for Java ME.[323] An internal Sun document stated: "Fragmentations was Java ME's biggest defect."[324] As a result, there were many different implementations of Java ME, and developers needed to engage in significant work to port an application written in the Java programming language from one Java ME implementation to another.[325] For example, an August 2006 presentation delivered internally by Shannon Lynch, Sun's then Senior Director of Mobile and Embedded Platforms, contains a slide on the existing fragmentation issue, and indicates that "[t]est and certification across device/platform/implementation combination drives 55% of development costs on average."[326] A September 2006 Sun presentation titled, "Java ME: Mobile and Embedded," also contains a slide with the heading, "With Fragmentation Further Undermining The Java Value Proposition,"

[321] "Sun Starts Bidding Adieu to Mobile-Specific Java," CNET, October 19, 2007, http://news.cnet.com/8301-13580_3-9800679-39.html?part=rss&subj=news&tag=2547-1_3-0-20, ("For Java ME, there are a large number of [specifications]… That posed a challenge to Java's original tagline, 'write once, run anywhere'…[W]ith the multiplicity of Java ME extensions, there was often little guarantee that a program written for one mobile phone would work on another"); Interview of John Rizzo; OAGOOGLE001208094.

[322] See, e.g., OAGOOGLE0100003277-291; OAGOOGLE0100005211-221.

[323] Brenner 12/15/2015 Dep., pp. 169:21-170:24.

[324] Hofert 12/1/2015 Dep., pp 153:15-154:16; Exhibit 1330 (OAGOOGLE2000077256-258 at 258).

[325] http://eriksdiary.blogspot.com; Interview of John Rizzo.

[326] OAGOOGLE0013331514-564 at 519.

and the bullet point: "'Having it your way' – liberal source distribution – accelerated adoption, but spawned implementation variations across the ecosystem."[327]

224. Sun's license with DoCoMo, a Japanese mobile phone operator, is a good example of how it created fragmentation. Under the license, DoCoMo implemented a proprietary version of Java ME known as DoJa (for "DoCoMo Java") that was incompatible with other implementations because it had different APIs.[328] This incompatibility meant that an application designed to run on the DoCoMo-specific DoJa platform would not run on Sun's Java ME platform.

225. Java ME was already fragmented as early as 2002—several years before Android development had even initiated.[329] Sun approved of this fragmentation as long as licensees tested their implementations using the Java Technology Compatibility Kit ("TCK") and paid a modest TCK fee to Sun. However, "passing" Sun's TCK ensures compatibility for only about 70% of an implementation.[330] This leaves substantial room for incompatibility among different Java ME implementations, with the end result of a highly fragmented Java ME marketplace. Another reason for the widespread incompatibility among implementations was that Sun allowed

[327] Brenner 12/15/2015 Dep., Exhibit 1395 (OAGOOGLE0011726508-539 at 511). See also OAGOOGLE0011761636-662 at 648; Brenner 12/15/2015 Dep., Exhibit 1390 (OAGOOGLE0013561757-786 at 779).

[328] Interview of John Rizzo.

[329] See various online sources included in Weingaertner Decl. in Support of Google Inc.'s Daubert Motion at ¶¶ 24-29, Exs. X-CC, June 14, 2011 (Dkt. No. 172).; Holzle 11/24/2015 Dep., p. 294:5-19

[330] Interview of John Rizzo; see also "James Gosling on Apple, Apache, Google, Oracle and the Future of Java," November 17, 2010 (video), http://www.youtube.com/watch?v=9ei-rbULWoA&feature=results_main&playnext=1&list=PL10A2B0EBC7523D48 (video of then Sun employee James Gosling).

"exceptions" to TCK failures.[331] Thus, some implementations of Java ME licensed and approved by Sun had not actually passed the TCK.

226. The result of this fragmentation was that the "write once, run anywhere" promise of the Java programming language had already been broken by Sun's own licensing practices well before Android's introduction.[332] It was not generally possible to write a Java language program for one handset and expect it to run on another handset.[333] A survey of developers concluded, "[o]ne could say that the vast majority of Java ME developers have lost faith in the write-once-run-anywhere vision."[334]

227. Fragmentation of Java ME was further increased by the very nature of the mobile handset space. With many different hardware configurations used in different models and by different manufacturers, the "write once, run anywhere" promise was going to be illusory in any event. As Mr. Schwartz testified, "the reality was unless and until [customers]

[331] Interview of John Rizzo.

[332] See, e.g., http://www.artima.com/forums/flat.jsp?forum=106&thread=205707. A blogger posted on May 16, 2007 that "[t]he main painpoint for Java ME is that it isn't 'write once, run anywhere'. Even apart from the different MIDP and CLDC versions, there are many differences in the various devices that support Java ME…In addition not all phones support all features correctly or completely. This makes it very difficult to create an interesting application (e.g., with some graphics) for a wide range of devices. You either have to develop for the lowest common denominator, or create different versions for different (groups of) devices. This is either very constraining, or very costly." Another blogger posted on the same date that "I'm currently in the middle of writing a JME application that should run on the top twenty phones in Europe, and I can tell you it is hell. It's like developing for 20 completely different operating systems."

[333] http://www.youtube.com/watch?v=9ei-rbULWoA&feature=results_main&playnext=1&list=PL10A2B0EBC7523D48, 9/25/2011 (video of then Sun employee James Gosling).

[334] "Making Sense of a Fragmented World: Mobile Developer Economics 2010 and Beyond, Insights and Analysis from the Definitive Mobile Developer Survey Plus Benchmarks on the Platform Development Experience," VisionMobile Ltd, July 2010, p. 14.

only wanted one phone, we were going to have to deal with trying to support a complex ecosystem."[335]

228. According to a document titled, "Mobile Java ME Issues and Thoughts on BRICA Strategy," "Implementation fragmentation" was one of the "Top pain points of mobile Java ME development." Furthermore, Java ME "[i]mplementations behave inconsistently due to bugs and varying behavior (stability, performance) – 'bad/unwanted' fragmentation." This fragmentation "kills the core value of Java ME (access to large numbers of devices and customers) and creates a huge barrier to entry especially for smaller ISVs and developers – which represent the vast majority."[336] According to an email from Terrence Barr of Sun, "The whole topic about how we are going to ensure the Java ME fragmentation and deployment story doesn't repeat on FX Mobile is something I am trying to understand better."[337] Also, according to Mr. Barr, prior to Android's release there was already "technical fragmentation" problems with Java ME.

> Q. …My question was, in a technical sense, Java ME was fragmented before Android was released, correct?
>
> THE WITNESS. I would like to qualify it and say that there was some technical fragmentation, yes.[338]
>
> ***
>
> Q. I want to go back to the point about fragmentation.

[335] Schwartz 7/20/2011 Dep., pp. 184:13-185:11.

[336] OAGOOGLE2000462635-638.

[337] OAGOOGLE0012917834-836. See also Barr 12/9/2015 Dep. Exhibit 1366.

[338] Barr 12/9/2015 Dep., p. 94:6-12.

> The fragmentation that we've been discussing throughout this deposition today is technical fragmentation of the Java ME platform, correct?
>
> A. Yes.
>
> Q. Is that fragmentation caused in any way by Android?
>
> THE WITNESS. Technical fragmentation of Java platform, no, that is not caused by Android.[339]

This technical fragmentation caused problems for Sun and contributed to the loss of developer mindshare.[340]

3. Sun Was Seeking a Partner to Extend the Life of the Java Platform

229. Sun's former CEO and President Jonathan Schwartz congratulated Google publicly after Google announced Android in 2007, saying that Google "just strapped another set of rockets to the community's momentum - and to the vision defining opportunity across our (and other) planets."[341] Sun perceived Android as an opportunity to revive the developer community's interest in Java, and create more business opportunities for Sun. Mr. Schwartz testified that "having Google embrace Java would be a positive for Sun, having them embrace our specific technologies would be economically positive, not simply PR positive."[342]

230. Mr. Schwartz testified that "[o]ver time, we only grew more interested in working with Google on Android because it was evident that smartphones would take over the marketplace and we didn't have a lot of confidence in our own ability to build an independent

[339] Barr 12/9/2015 Dep., pp. 279:23-280:8.

[340] Barr 12/9/2015 Dep., pp. 248:13-249:21, 280:11-281:12.

[341] Schwartz 7/20/2011 Dep., pp. 113:10-114:22, 118:08-122:16; Schwartz 7/20/2011 Dep., Exhibit 66 (GOOGLE-00-00000512).

[342] Schwartz 7/20/2011 Dep., pp. 106:5-107:4.

handset, and Google appeared through their alliances as well as their technology work to be best positioned to really drive a device that would carry the Java programming language to that next generation of mobile handset users."[343]

231. Sun recognized that working with a partner or partners like Google would give its Java platform a better chance to succeed in the mobile space and to adapt to the next generation of mobile devices.[344] Partnering with Google on Android would have provided such a valuable opportunity for Sun that Sun would have partnered up with Google "for free" or "might have even paid for the privilege to participate," as long as Google elected to use Java.[345] This is "[b]ecause the net effect of being engaged with a handset platform that we knew would be globally successful would have a considerable halo effect for Sun just in our own brand image, leaving aside access to carriers to talk about how we could create, you know, alternatives and variations for them."[346]

232. Mr. Schwartz further explained that:

> It would have been of significant strategic value for Sun to be on stage with Google announcing Android. That had value, potentially significant value. To not be on stage was not as valuable. So would the 10 million, which if I recall was year one payment from Google to Sun, would that have changed in any way our financial destiny? No. So would we have waived that for ultimately being a premier technology partner alongside Google? I believe we would have. I didn't have that option.[347]

[343] Schwartz 7/20/2011 Dep., p. 157:15-23. See also Gupta 7/26/2011 Dep., pp. 119:06-122:06.

[344] OAGOOGLE0018885324.

[345] Schwartz 7/20/2011 Dep., pp. 110:8-112:1.

[346] Id.

[347] Schwartz 7/20/2011 Dep., pp. 112:08-113:3.

4. Other Factors Having Nothing to Do With Android Contributed to Java ME Revenue Losses

233. Java ME faced a number of challenges having nothing to do with Android that would be expected to have caused revenues to decrease over time even in the absence of Android, including among others, its own decision to open source Java ME, Sun's lack of investment in Java ME, the move in the mobile space toward smartphones, the financial crisis, and a history of security problems with Java and Oracle's failure to address these problems promptly.

a. Open Sourcing Java ME

234. In December 2006, Sun made the Java ME platform available through an open source license.[348] Open source had become an industry trend, and because Sun was not open sourcing, the result was "we were finding developers less interested in engaging with us, using our technology, and as a result, we were finding fewer customers interested in buying from us."[349] Sun's former CEO Jonathan Schwartz testified that Sun opted for making the open source distribution available as "a competitive move," and that "we had no choice…it was evident, given the rise and the very pervasive success of Linux on servers, that the same thing would happen on handsets."[350] Under the open source license, users can use the Java ME platform for free unless they want to call their system "Java."[351] By 2008, the only part of the entire Java platform that was not made open source consisted of a few mobile extensions of

[348] Schwartz 7/20/2011 Dep., pp. 24:5-24, 27:9-28:17.

[349] Schwartz 7/20/2011 Dep., pp. 64:20-65:10.

[350] Schwartz 7/20/2011 Dep., pp. 32:17-22, 35:4-25.

[351] Schwartz 7/20/2011 Dep., pp. 35:21-25, 46:12-50:17.

JavaFX.[352] By open sourcing, Sun was hoping to attract a larger developer community, and to obtain revenue from those customers seeking to be certified with the "Java" name, and pay for other "derivatives", i.e., commercial support services.[353]

235. Sun's strategic forecasts from late 2007/early 2008—the same forecasts used by Mr. Malackowski to calculate his "but-for" Java ME licensing revenue in his calculation of Oracle's lost profits from Java ME licensing—predicted a decline in Java ME licensing revenues due to Sun's decision to open source for fiscal years 2009 and 2010. Specifically, the Low ("Major shift to open source"), Medium ("Moderate shift to open source"), and High (Minor shift to open source") forecast scenarios all forecast a decline in Java ME licensing revenues apparently due to open sourcing Java ME. I note that Mr. Malackowski ignores these three scenarios, and, based solely on discussions with Michael Ringhofer, VP of Worldwide Java Business at Oracle, relies only on the Strategic Forecast scenario because he believes "this projection is most in line with the business strategy given what was known at the time."[354] To simply pick the Strategic Forecast scenario, which appears to assume no effect on Java ME licensing revenue due to open sourcing, and which happens to be the scenario with the largest Java ME licensing revenues and therefore generates the largest lost profits damages for Mr. Malackowski, makes no economic sense. It is notable that these Sun forecasts make no mention of the effect of Android on Sun's Java ME licensing revenues, or any of Sun's Java licensing businesses.

[352] Schwartz 7/20/2011 Dep., pp. 57:02-60:24; Schwartz 7/20/2011 Dep., Exhibit 54 (OAGOOGLE0003901182).

[353] Schwartz 7/20/2011 Dep., pp. 69:14-72:7.

[354] Malackowski 1/8/2016 Report, ¶ 186.

236. Furthermore, former Sun employee Eric Chu testified in response to the question—"Was there any impact on Java ME licensing revenues from the open source version of Java ME"—that open sourcing did contribute to at least some of the decline in Java ME licensing revenues.[355] Alan Brenner, former Vice President of Engineering for Java Card and Java ME, stated the following with regards to the effect on Java ME licensing revenues due to Sun open sourcing Java ME under its proposed collaboration with Google: "To say it another way, the Java ME business at this point in time was generating somewhere in the vicinity of two-to $300 million in revenue a year. Open-sourcing of Java ME would relatively rapidly negatively impact that business, and so I believed that this was one of several ways in which Sun was discussing with Google to compensate Sun for that business impact."[356]

b. Sun's Lack of Investment in Java ME

237. Mr. Chu also testified that Sun's lack of investment in Java ME and the complicated licensing structure of Java ME contributed to the decline in Java ME licensing revenues.

> It is the lack of investment from Sun in Java ME. It's the fact that companies like Nokia, like Siemens, and others, also start building specs with TCKs that now everybody has to go license, and so if you want to build a handset that includes Java ME technologies, now you have to go through Sun and Nokia and Siemens and Motorola to license different pieces. That's what has caused a lot of problems.[357]

Mr. Chu further testified:

> Yeah, and I have one more thing to the last comment.

[355] Chu 4/28/2011 Dep., pp. 160:18-161:11.

[356] Brenner 12/15/2015 Dep., pp. 197:20-198:1.

[357] Chu 4/28/2011 Dep., p. 161:17-162:4.

> In fact, Sun was really losing control of Java ME already at that time because more and more of the Java ME platform was defined by somebody else and a TCK, which basically defined compatibility, had to be licensed from somebody else. And I think that's an important issue that caused a lot of pressure on the revenue.
>
> If Sun was not investing and adding new capability and technologies, everybody else were. And so it's – Sun had control of a very small piece of all the pieces that called Java ME. Everybody else had control of all the other pieces. [358]

238. Alan Brenner gave a Sun presentation that listed, "Too little Sun marketing to continue to promote Java ME in ecosystem," as a reason for the "Decrease in Relevance of Java ME."[359] Additionally, around 2013, Sun decided "not to focus on making a new major version of Java ME update, targeting the phone business, because we didn't believe that we would get a net option for it, basically."[360]

239. Terrence Barr testified about Sun's failure to invest in Java ME.

> Q. Okay. What other – for what other reasons did you believe that the mobile Java strategy at Sun was failing at that time?
>
> A. The primary reasons – and I believe we discussed those already – were that I believe the mobile Java platform needed to evolve to encompass more functionality and provide developers with more features to develop applications, and so that was my primary concern, that – that we needed to address that and evolve the Java platform, the mobile Java platform.[361]

Henrik Stahl, VP of Product Management at Oracle, also testified about Sun's failure to invest in Java ME.

---

[359] Brenner 12/15/2015 Dep., Exhibit 1390 (OAGOOGLE0013561757-786 at 781).

[360] Stahl 1/14/2016 Dep., p. 230:20-24. See also Stahl 1/14/2016 Dep., pp. 318:19-319:5. ("Oracle made a decision to not invest in a – well, there were two decisions made. One was don't invest in an updated version of Java ME for low-end phones. Instead let the next evolution of Java ME be focused only on small embedded.").

[361] Barr 12/9/2015 Dep., p. 158:8-18. See also Barr 12/9/2015 Dep., pp. 159:6-25, 161:15-162:16, 165:1-22.

Q. Was it your view, at the time of this document in April 2012, that Java ME was an old technology stack?

A. Yes, I believe that if you wanted to continue to use and license Java ME and, in particular, to be able to compete with something like Android, would have to make significant investments in it.

Now, that might not be the right decision, which is kind of what the third bullet here is referring to.

Q. And Oracle did not, in fact, make the significant investment required to keep up, correct?

A. We considered it.

A. We considered it. We decided against it.[362]

Mr. Stahl further testified that Oracle would have to incur significant costs to improve the Java ME platform to the point that it was competitive in the modern smartphone market.

Q. Do you believe, as you sit here today, that Oracle would have to incur significant costs to create a mobile phone software stack?

A. I believe that creating a mobile phone software stack is a significant investment, yes. So for Oracle, it would be a significant investment to do that, which would have to be offset by, you know, revenue in some shape, way or form.
…
Q. So it takes a significant amount of time and resources to build a mobile phone software stack?

A. That's my layman estimate, yes.

Q. And Google undertook that time and expenditure of resources to build Android, correct?

A. Yes, I believe so.

Q. And Oracle chose not to undertake the time and resource expenditure to build its own competing software stack for mobile phones, correct?

[362] Stahl 1/14/2016 Dep., p. 208:6-24.

A. So we are now looking at the time frame, late 2011 to 2012, yes, Oracle intentionally decided not to do that. That's correct.[363]

***

Q. Has Oracle ever had a product that could compete in the smartphone market?

A. As far as I'm aware, no, nor have I been aware of any attempt to build such a stack."[364]

c. The Move in the Mobile Space towards Smartphones

240. Another contributing factor to the decline in Java ME revenues was that Java ME "was developed for feature phones," and "was not fully capable for what is required by a smartphone device."[365] It was designed to take up a very small amount of memory and to work with slow and limited functionality microprocessors.[366] A Sun document states, "Java ME revenue under pressure because feature phone…is declining overall and relative to smartphones."[367] Another Sun document titled "Java Strategy Update," described Java ME as "[p]ositioned for feature phones and 'dumb' phones."[368] The same document stated: "Samsung and Nokia only remaining feature phone vendors, plus long tail of tier 3 OEMs in China and India…Expect decreasing revenues for 5 years, then little more business."[369] Michael

[363] Stahl 1/14/2016 Dep., pp. 173:12-21,174:18-175:10.

[364] Stahl 1/14/2016 Dep., pp. 153:23-24, 154:1-3.

[365] Rizvi 7/28/2011 Dep., p. 203:5-7, 17-19. See also Hofert 12/1/2015 Dep., p. 92:14-21. ("Q. And then Java ME says that it's for devices. That's a broad category. But what would fall within the category of devices, again, from the perspective of approximately 2007 when this document was created? A. In 2007, this would refer to feature phones, it would refer to smaller printers, some handhled devices. Things like that.")

[366] Schwartz 7/20/2011 Dep., p. 25:13-18.

[367] "Java Market Opportunities," Jeet Kaul, OAGOOGLE0100164986-016 at 999.

[368] Ringhofer 12/11/2015 Dep., Exhibit 1346 (OAGOOGLE2000181018-074 at 027).

[369] Ringhofer 12/11/2015 Dep., Exhibit 1346 (OAGOOGLE2000181018-074 at 027).

Ringhofer, VP of Worldwide Java Business at Oracle, is not aware of any Oracle efforts since 2011 to license Java ME for use in smartphones other than BlackBerry.

> Q. Other than the BlackBerry phone that you mentioned, are you aware of any effort by Oracle to license Java for use in a smartphone since 2011?
>
> A. I am not aware of any specific model that I can tell you now, no.[370]

241. Mr. Ringhofer also stated that the type of phone that Oracle typically licensed Java ME has been in decline since 2011, and the decline for this type of phone has been a reason for the decline in Java ME licensing revenues.

> Q. Is it fair to say that the type of phone for which Oracle typically licensed Java ME is losing market share since 2011?
>
> A. Yes.
>
> Q. There are less of those types of phones now than were in 2011?
>
> A. Yes.
>
> Q. And is that the reason Oracle's Java ME licensing revenues have gone down?
>
> A. I don't believe that's the only reason.
>
> Q. Is it a reason?
>
> A. Sure.
>
> Q. I don't want to put words in your mouth –
>
> A. Yes, that is a reason.[371]

[370] Ringhofer 12/11/2015 Dep., p. 273:17-24.

[371] Ringhofer 12/11/2015 Dep., p. 395:1-24. See also Stahl 1/14/2016 Dep., pp. 207:23-208:1. ("They moved away from low-end phones where Java ME was the best solution to run Java applications, which is, I guess, what we mean when we say feature phones here.")

242. Furthermore, a Sun Java ME presentation stated that, "Java ME losing Business Relevance in mobile."[372] Alan Brenner, former Vice President of Engineering for Java Card and Java ME, expanded on the reasons for "the decrease in relevance of Java ME": "I – well, as I've tried to say several times, what was meant by that statement is the landscape was changing, more capable devices were emerging, and the industry was consolidating. And to renew the value of our product, we needed to respond to those changes by providing additional implementation depth, and this presentation captures some of the things you need to do to do that."[373]

243. After the time that Sun prepared one of its forecasts of Java ME licensing revenues (i.e., the late 2007/early 2008 forecasts used by Mr. Malackowski), the trend was toward smartphones and away from feature phones. Worldwide smartphone shipments increased from 119.56 million units in 2007 to 146 million units in 2008 and 169.98 million units in 2009.[374] This trend would have occurred even without Android. Indeed, Android's shipments were only 0.5 million units in 2008 and 6.97 million units in 2009 worldwide.[375]

244. Given that Java ME was highly dependent on feature phones, the shift away from feature phones toward smartphones accounted for at least part of the decrease in Java ME licensing revenue, and Android smartphones represented only a part of this shift because Android had only a share of the worldwide smartphone market. In a world without Android, many of the Java ME sales on feature phones that Mr. Malackowski claims were lost to Android

[372] Brenner 12/15/2015 Dep., Exhibit 1390 (OAGOOGLE0013561757-786 at 769).

[373] Brenner 12/15/2015 Dep., pp. 158:8-18.

[374] "Global Smartphone Sales Forecast by Operating System and Region," Strategy Analytics, January 2011.

[375] "Global Smartphone Sales Forecast by Operating System and Region," Strategy Analytics, January 2011.

would have been lost to other smartphone platforms, such as the iPhone, instead.[376] Since Apple does not use Java ME on the iPhone and Java ME is not well-suited for other smartphones, it is likely that Java ME revenue would have decreased even in the absence of Android.[377]

245. It is likely that Sun's Java ME licensing revenues forecast prepared in late 2007/early 2008, and used by Mr. Malackowski to calculate lost profits, understated the effect that the shift away from feature phones toward smartphones, and in particular the effect of the iPhone, would have on Sun's Java ME licensing revenues. Third-party forecasts for the iPhone prepared around the time of Sun's Java ME licensing forecasts used by Mr. Malackowski generally under-forecasted the demand for the iPhone. For example, actual sales of the iPhone exceeded analyst forecasts by 46% over the 2008-2009 period.[378]

246. As a result, Sun's forecasts of Java ME licensing revenues as of late 2007/early 2008 likely overstate the actual Java ME licensing revenues that Sun could have actually achieved, even without Android in the market, in light of the general under-forecasting of demand for smartphones (in general) and the iPhone (in particular). This means that Mr. Malackowski's calculation of lost profits based on Sun's forecasts of Java ME licensing revenues, which misrepresent what Sun could have actually achieved even without Android in the market, is unreliable and inappropriate for purposes of a lost profits calculation.

---

[376] As discussed above, the iPhone competes closely with Android handsets.

[377] OAGOOGLE0018885324 ("Developers are moving from Java to Apple and Android").

[378] Actual iPhone unit volumes from "WW Quarterly Mobile Phone Tracker," IDC, November 13, 2015. Analyst forecasts from UBS Investment Research, Apple Inc., "Macs Humming into Macworld," December 17, 2007, p. 10; Morgan Stanley, Apple Inc., "Shifting Focus to New Products and Margins," p 7; Bear Stearns, Apple Inc., "Raising Estimates and CY08 Target to $249 on Favorable Retail Survey and Asia Feedback," pp. 3-4. Data is adjusted for Apple's fiscal year ending September 30.

247. I note that Mr. Malackowski states that Sun's forecasts to some extent were influenced by the threat of Android. He cites to a December 10, 2007 Sun document that states "Java ME under attack" and goes on to state that "the most concerning of all is the combination of Android's Dalvik VM + Linux" and "given their resources, Google will outspend and underprice us."[379] Mr. Malackowski's interpretation of this document is contradicted by both Sun and third party analysts. Specifically, according to former Sun employee, Vineet Gupta, upon Google's release of Android in late 2008, Sun did not view Android as a threat: "Maybe it was 2008. A lot of us had internal discussion on – to ensure what our customers want and need. And I'll be very honest with you, a lot of our customers did not expect Android would work, and they wanted to continue working with Java. So we didn't see it as a threat at all at that time."[380] Mr. Gupta's opinion is particularly significant because, in 2005 and 2006, he had been Sun's lead negotiator in its discussions with Google about a partnership to develop a Java-based mobile platform. In that role, Mr. Gupta was tasked with understanding and, if possible, quantifying the competitive threat, if any, posed to Sun's Java licensing business by an open-source Java-based mobile platform, so that Sun could take account of those threats in its negotiating position. Additionally, in late 2008, the market expressed the same sort of doubts Mr. Gupta expressed regarding Android's chances of success: "no one is expecting Android to be a major success overnight. The mobile market is becoming very crowded, with Microsoft, Nokia and others also fighting for smart phone market share. Market research firm J Gold Associates estimates Google/Android market share will reach about 5% in three years.

[379] Malackowski 1/8/2016 Report, ¶ 188, quoting OAGOOGLE0009784791-800 at 795.

[380] Gupta 7/26/2011 Dep., p. 136:8-13.

Research firm Gartner is slightly more optimistic, predicting a 10% share in the same time frame. Piper Jaffrey analyst Gene Munster said he doesn't expect Android to overtake the iPhone anytime soon."[381]

d. The 2008 Financial Crisis

248. The financial crisis of 2008, which was a cause of the failure of JavaFX Mobile as well as a decline in Sun's revenues generally, likely was another contributory factor to the decrease in Java ME licensing revenues that had nothing to do with Android.[382]

249. From 2007 to 2008, the S&P 500 Index decreased by 38.5%.[383] According to Sun's annual report around that time, "The financial crisis in the U.S. and global economic recessions has negatively affected our business, results of operations, and financial condition, and could continue to do so in the future."[384] I note that Sun's revenues for its Systems Group, which includes software revenues from licensing Java products, decreased by -1.7% in FY 2008 ($8,771 million to $8,618 million), and by -22.2% in FY 2009 ($8,618 million to $6,704 million).[385] It is necessary to account for the impact that the 2008 financial crisis would have had on the likelihood that Sun could achieve its sales forecasts for its Java ME licensing revenues.

---

[381] "Google's Mobile Ambitions; No Overnight Success," Richard Karpinski, AdAge, October 13, 2008, http://adage.com/article/btob/google-s-mobile-ambitions/273071/.

[382] Schwartz 7/20/11 Dep., pp. 60:21-61:7 ("a burgeoning financial crisis that…saw our top line decrease by 30-plus percent").

[383] "S&P 500 Historical Prices December 31, 2007-December 21, 2008," http://finance.yahoo.com/q/hp?s=%5EGSPC&a=11&b=31&c=2007&d=11&e=31&f=2008&g=d&z=66&y=198.

[384] Sun Microsystems 10-K, 2009 Annual Report, June 30, 2009, p. 12.

[385] Id. at p. 90.

e. A History of Security Problems with Java and Oracle's Failure to Address These Problems Promptly

250. There has been a history of security problems with Java, and Oracle has often failed to address these problems promptly.

251. For example, in January 2010, the Java zero-day bug affected all versions of Java 7.[386] In late 2011/early 2012, Operation Red October was "a massive espionage malware campaign that went undetected for five years relied in part on a vulnerability in the widely deployed Java software framework to ensnare their victims."[387] In January 2013, "[s]ecurity researchers have confirmed that the latest version of Oracle's Java software is vulnerable to Web hacks that allow attackers to install malware on end users' computers."[388] According to Adam Gowdiak, CEO of Security Explorations, "We have successfully confirmed that a complete Java security sandbox bypass can be still gained under the recent version of Java 7 Update 11 (JRE version 1.7.0_11-b21) … As a result, two new security vulnerabilities (51 and 52) were spotted in a recent version of Java SE 7 code and they were reported to Oracle today (along with a working Proof of Concept code)."[389] Mr. Gowdiak's advisory was issued a few days after security firms, Trend Micro and Immunity Inc., both independently reported that the emergency patch recently released by Oracle "was incomplete."[390] Around the same time,

[386] "Critical Java zero-day bug is being 'massively exploited in the wild' (Updated)" arstechnica, January 10, 2013, http://arstechnica.com/security/2013/01/critical-java-zero-day-bug-is-being-massively-exploited-in-the-wild/.

[387] "Red October relied on Java exploit to infect PCs," arstechnica, January 14, 2013, http://arstechnica.com/security/2013/01/massive-espionage-malware-relied-on-java-exploit-to-infect-pcs/.

[388] "Critical Java vulnerabilities confirmed in latest version," arstechnica, January 18, 2013, http://arstechnica.com/security/2013/01/critical-java-vulnerabilies-confirmed-in-latest-version/.

[389] Id.

[390] Id.

"[s]ecurity researchers have uncovered a newly discovered bug in Oracle's Java framework that allows attackers to bypass important security protections designed to prevent malware attacks."[391] As a result, "many security professionals have called on Oracle to communicate more quickly and effectively when it learns of new vulnerabilities in recent version of its software."[392]

252. In February 2013, "Twitter engineers shut down what they described as an 'extremely sophisticated' hack attack on its network that exposed the cryptographically protected password data and login tokens for 250,000 users."[393] As a result, "Twitter urged users to disable Java on their computers."[394] Around the same time, "Facebook officials said they recently discovered that computers belonging to several of its engineers had been hacked using a zero-day Java attack that installed a collection of previously unseen malware."[395] Apple and Microsoft also experienced similar problems associated with Java security problems.[396]

253. The Department of Homeland Security has also experienced Java-related security issues and advised computer users generally to disable Java.

---

[391] "Java's new 'very high' security mode can't protect you from malware," arstechnica, January 28, 2013, http://arstechnica.com/security/2013/01/javas-new-very-high-security-mode-cant-protect-you-from-malware/.

[392] Id.

[393] "Twitter detects and shuts down password data hack in progress," arstechnica, February 1, 2013, http://arstechnica.com/security/2013/02/twitter-detects-and-shuts-down-password-data-hack-in-progress/.

[394] Id.

[395] "Facebook computers compromised by zero-day Java exploit," Ars Technica, February 1, 2013, http://arstechnica.com/security/2013/02/facebook-computers-compromised-by-zero-day-java-exploit/.

[396] "Apple HQ also targeted by hackers, will release tool to protect customers," Ars Technica, February 19, 2013, http://arstechnica.com/apple/2013/02/apple-hq-also-targeted-by-hackers-will-release-tool-to-protect-customers/; "Microsoft joins Apple, Facebook, and Twitter; comes out as hack victim," Ars Technica, February 22, 2013, http://arstechnica.com/security/2013/02/microsoft-joins-apple-facebook-and-twitter-comes-out-as-hack-victim/.

> Despite an emergency software update issued yesterday by Oracle, the U.S. Department of Homeland Security is still advising computer users to disable Java on their Web browsers, fearing that an unpatched vulnerability remains.
>
> Oracle released a software update on Sunday to address a critical vulnerability in Oracle's Java 7 after the DHS' Computer Emergency Readiness Team issued an advisory last week recommending users disable the cross-platform plugin on systems where it was installed. The flaw could allow a remote, unauthenticated attacker to execute arbitrary code when a vulnerable computer visits a Web site that hosts malicious code designed to take advantage of the hole.
>
> Oracle said in an advisory yesterday that it "strongly" recommended users update their Java software to repair the vulnerability. But the DHS is still worried that further, unknown flaws may exist in Java.
>
> "This and previous Java vulnerabilities have been widely targeted by attackers, and new Java vulnerabilities are likely to be discovered," DHS said in an updated alert published on the CERT Web site. "To defend against this and future Java vulnerabilities, consider disabling Java in Web browsers until adequate updates are available."
>
> Security company Immunity reported that Oracle's update addressed only one vulnerability and that another still existed.
>
> "The patch did stop the exploit, fixing one of its components," Immunity said in a blog post today. "But an attacker with enough knowledge of the Java code base and the help of another zero day bug to replace the one fixed can easily continue compromising users."[397]

254. Apple, in particular, has shut down Java multiple times due to security issues. For example, in October 2012, "Apple has recently pulled Java from OS X in an effort to close

[397] "Homeland Security still advises disabling Java, even after update," CNET, January 14, 2013, http://www.cnet.com/news/homeland-security-still-advises-disabling-java-even-after-update/ (emphasis in original).

some of the loopholes that potential attackers could use to compromise a Mac."[398] In August 2013, "[f]or at least the third time this year, Apple has **blocked** the Java 6 and Java 7 plug-ins on Macs due to 'multiple security issues' in versions older than the most current."[399] "The security problems surrounding Java have become so serious that Apple was forced to take the unprecedented step of **remotely disabling** the plug-in to prevent the spread of malware and identity theft late last year, and have periodically had to re-block Java as new and serious security issues were uncovered. Java use on the Internet has dropped dramatically as a result of the issues, but many web sites still have Java-based web apps that can run in order to allow both Mac and Windows users access to the same programs."[400]

255. As discussed above, Oracle has been slow to address Java security flaws. For example, in April 2012, Security Explorations disclosed details to Oracle on 31 Java security issues. Of those 31 issues, only 2 had been fixed in the Java Critical Patch Update that was released in June 2012. According to Mr. Gowdiak, "We…expected that the most serious of them would be fixed by June 2012 Java CPU…But it didn't happen and Oracle left many issues unpatched with plans to address them in the next Java CPUs."[401]

[398] "Apple Drops Java in Latest OS X Security Release," The Mac Security Blog, October 23, 2012, https://www.intego.com/mac-security-blog/apple-drops-java-in-latest-os-x-security-release/.

[399] "Apple blocks Java plug-in once again over security issues," MacNN, August 29, 2013, https://www.macnn.com/articles/13/08/29/keeping.java.up.to.date.can.help.avoid.inconvenience/.

[400] Id.

[401] "Oracle knew about critical Java flaws since April," The Register, August 30, 2012, http://www.theregister.co.uk/2012/08/30/oracle_knew_about_flaws/.

f. Summary

256. As discussed in the previous sections, Java ME licensing revenues declined for a number of reasons. Furthermore, Java ME licensing revenues would have still declined for these reasons even in the absence of Android. As a result, it would be inappropriate to assign the whole decline in expected Java ME licensing revenues (compared to actual Java ME licensing revenues) to Android, as Mr. Malackowski has done. Moreover, this is without considering that Android would still have existed in the but-for world.

5. Sun's Attempt to Enter with JavaFX Mobile/Project Acadia and Oracle's Subsequent Attempts to Create a Java Phone Were Unsuccessful for Reasons Unrelated to Android

257. Sun acquired SavaJe in April 2007, hoping that SavaJe would provide it with "a vertically focused solution that would allow you to provide everything you needed to—or most of the things you need to bring a phone to market."[402] After the acquisition, Sun referred to the initiative of bringing a full software stack to the market under the name of Project Acadia or JavaFX mobile.[403] Sun's former CEO, Jonathan Schwartz described JavaFX Mobile as Sun's plan to "create a slightly larger phone form factor implementation that would give you access to the internet and [other functions]."[404]

258. Project Acadia received mixed reactions from handset manufacturers such as Nokia, Motorola, SEMC, Samsung, and LG. From the very beginning (April 2007), handset manufacturers expressed skepticism about the level of production, and questioned why Sun

[402] Gering 7/20/2011 Dep., pp. 128:14-24.

[403] Gering 7/20/2011 Dep., pp. 132:5-133:4; Singh 6/23/11 Dep., pp. 109:13-110:7, 116:1-117:10; Schwartz 7/20/2011 Dep., p. 53:4-13. The name "JavaFX mobile" was also used to refer to the UI presentation layer set of technologies at a later date.

[404] Schwartz 7/20/11 Dep., p. 128:9-22.

would succeed where SavaJe had failed despite making a significant investment.[405] One discussion also pointed to the major issue with SavaJe: "due to its high requirements for CPU and memory usage the performance of the stack wasn't very good. If Sun is to be successful, this *must* be managed effectively and a suitable chipset chosen."[406]

259. After the acquisition of Savaje, "doing a more complete handset stack became [Sun's] real strategy of record a couple of years ago. This died nearly a year ago [approximately May 2008] partly because [Sun] couldn't fund it, and partly because the carriers & manufacturers didn't really want it…We had a pretty big handset applications team, but they disappeared in the last RIF…It would be a wonderful thing to do, almost certainly very lucrative, but the upfront NRE is a killer."[407] Sun had several rounds of "workforce reductions" since 2002, including 2,150 in May 2006, 1,450 in August 2007, 1,500-2,500 in May 2008, and 5,000-6,000 in November 2008—all before, or only a month after in the case of the November 2008 reduction, Android's introduction.[408]

260. Mr. Schwartz testified that Sun did not have a lot of confidence in their own ability to build an independent handset platform because that would require support from carriers, handset manufacturers, and "a more robust…technology platform," all of which Sun was lacking.[409] Indeed, Sun had never previously built a commercial implementation for the

[405] OAGOOGLE0001342929-31.

[406] OAGOOGLE0001342929-34.

[407] An email string between James Gosling and other Sun/Oracle employees in May 2009. Trial Exhibit 2043 (OAG00GLE0007622843).

[408] OAGOOGLE0100072599 at 602.

[409] Schwartz 7/20/2011 Dep., pp. 158:8-159:2.

mobile space.[410] Instead, Sun had used a licensing model. Thus, it was essentially an entirely new business model that Sun was considering, one with which it had no experience.

261. The reason JavaFX Mobile failed to monetize was "[p]artially because [Sun] ended up being focused on other things and JavaFX Mobile never shipped as a complete software platform," as well as the stress from the financial crisis.[411] Sun had other priorities and did not provide enough investment and focus on Project Acadia. As former Sun CEO and President Jonathan Schwartz testified, "It has a propensity -- as a 12, $13 billion a year company to focus on much bigger items than whether a -- you know, million-dollar product line is going to be Open Sourced or not."[412]

262. Various Sun Executives made statements such as "we should just give FX mobile away…;" "There seems to be more interest in silo protection than in solving the problem…There are significant gaps in our technical leadership, especially with mobile & embedded experience."[413] Former Sun/Oracle employee Craig Gering testified that there were communication issues between two teams responsible for developing JavaFX mobile.[414]

263. Although Sun was not successful, it was not for lack of opportunity in the smartphone market for an additional platform. If Sun had the capability to take effective measures to mitigate its damages, it could have become a competing smartphone platform supplier. For example, some handset manufacturers that are existing Android users such as

[410] Interview of John Rizzo.

[411] Schwartz 7/20/2011 Dep., pp. 60:22-24, 152:25-153:4.

[412] Schwartz 7/20/2011 Dep., p. 61:4-7.

[413] Gering 7/20/2011 Dep., Exhibit 104 (OAGOOGLE0000478601).

[414] Gering 7/20/2011 Dep., pp. 142:24-144:4.

HTC, Samsung, LGE, and Motorola, started to hedge themselves against being overly dependent on Android by investing in devices that support other platforms such as BREW, Microsoft, Meego, and Bada.[415]

264. Similar to Sun's failure to deliver JavaFX, Oracle also considered and failed to deliver a Java Phone to the market. Oracle ultimately decided that it was not a good idea to go forward with the Java Phone project. In particular, various Oracle executives and representatives testified that Oracle had thought about developing a Java Phone, and eventually concluded that it was a "bad idea."[416] Indeed, Mr. Ellison, Oracle's CEO and Head of Engineering, testified that, "It was an idea that perhaps a good way or the best way would be to build a Java phone. I had the idea for building a Java phone. And we explored that idea, decided it would be a bad idea."[417]

265. As was the case with Sun, there were many conditions, other than the presence of Android, which led to Oracle abandoning the development of the Java Phone. For example, Oracle's lack of capacity and competency in developing smart phone software was one of them.[418] Mr. Ellison testified that Oracle might have to buy some companies or hire "a bunch of people from outside," since Oracle did not have the capability in-house to build the Java smart phone software.[419] Indeed, Oracle even considered buying Palm in order to acquire the

[415] See, e.g., "Android: On a Bender, Telecom Equipment," Arete Research Services LLP, July 14, 2010, GOOGLE 01-00049780-784 at 780.

[416] Trial Tr. 318:17-20, 320:19-22 (Ellison).

[417] Trial Tr. 318:17-20 (Ellison).

[418] Trial Tr. 331:3-6 (Ellison).

[419] Trial Tr. 331:9-12 (Ellison).

necessary technology and expertise.[420] Mr. Ellison further testified at trial: "We didn't have -- we had never built a smart phone before. That wasn't Oracle's business. Oracle's Java business was on big server computers, you know. We have never built a smart phone before. We had no experience in-house on this at all, or very little."[421]

266. Henrik Stahl, VP of Product Management at Oracle, testified in his deposition that he is not aware of any Oracle efforts to "build and sell" a phone.

> Q. Are you aware of any effort by Oracle to create a Java-based smartphone?
>
> A. The definition of smartphone is a bit vague, but I'm not aware of any effort by Oracle to create any phone.
>
> Q. So you are not aware of any effort by Oracle to create any type of Java-based phone?
>
> A. If by that you mean build and sell a physical phone, no.[422]

***

> Q. Has Oracle ever made any attempts to build a complete phone software stack?
>
> A. To the best of my knowledge, no."[423]

Mr. Stahl further testified that Oracle did not make an effort to stay informed of the evolving mobile phone technology.

> Q. Has Oracle made any effort to keep up with the evolving technology for mobile phones?
>
> A. That's a variation of a question you asked earlier. We are not investing in phones. We made a conscious decision in the, you know, 2011, 2012 time frame not to invest in

[420] Trial Tr. 340:4-7 (Ellison)("As you said earlier, we didn't have a lot of -- we had no people in-house that had ever built a smart phone before. One way to get those people in-house would have been to buy Palm.").

[421] Trial Tr. 338:20-24 (Ellison).

[422] Stahl 1/14/2016 Dep., p. 131:10-19.

[423] Stahl 1/14/2016 Dep., p. 153:13-15.

> software for phones, except opportunistically if we had a large customer asking for something specific. Those decisions were based on the simple observation that we didn't think there was any market left after Android appeared and got dominance, and no investments have been made since, as far as I'm aware.[424]

267. Furthermore, Mr. Stahl also testified that Oracle did not actively pursue either the low-end or high-end of the phone market in general.

> It's correct that Oracle actively decided against pursuing the phone market. Their main reason for which was that we didn't believe there was enough revenue in it for us, given that high-end phones would use Android, which they would get from Google for free, and it's very hard to compete with free. And low-end phones or lower-end phones, whether they were running Android or something else, would have too small margins for the producers of such phones to be willing to pay us any, you know, reasonable price for our technology."[425]

268. In summary, given that Sun's Project Acadia/JavaFX Mobile failed in the marketplace, and Oracle was also unable to successfully deliver a Java Phone to the market, and that there were many reasons for these failures that had nothing to do with Android or Google's alleged infringement, it is speculative to assume that Project Acadia/JavaFX Mobile or a Java Phone would have succeeded in the absence of Android. Relatedly, Mr. Malackowski states that "[t]he amount of losses attributable to the loss of the Acadia platform is very difficult to quantify since the product never achieved distribution agreements nor launched."[426] Thus, in the end, Mr. Malackowski does not put forward any damages figure related to Project Acadia.

269. However, I note that Mr. Malackowski does attempt to equate the potential Project Acadia losses to Google's alleged unjust enrichment: "Therefore, I believe Sun and later

[424] Stahl 1/14/2016 Dep., pp. 162:19-163:7.

[425] Stahl 1/14/2016 Dep., p. 159:13-25.

[426] Malackowski 1/8/2016 Report, ¶ 215.

Oracle's actual losses attributable to the lost Acadia opportunity could be quite significant, and, potentially best measure by the apportioned Android profits attributable to the Infringed Java Copyrights. In other words, Google's Android-related profits represent, in some part, Sun and Oracle's inability to pursue the exact same market opportunity for a Linux/Java SE based smartphone because Google was competing against them using their own Java Copyrights."[427] First, Mr. Malackowski undertakes no quantitative or qualitative analysis to support this statement. Second, Mr. Malackowski's statement implies, based on his own analysis of Google's alleged unjust enrichment (before apportionment to the allegedly infringing material), that Oracle's actual losses due to Project Acadia are in the billions of dollars. This amount is inconsistent with the calculation of actual losses due to Project Acadia calculated by Oracle's previous damages expert, Dr. Cockburn, of only $17.9 million.[428] Mr. Malackowski offers no explanation for why he believes Oracle's actual losses due to Project Acadia are substantially higher than those claimed by Oracle's previous expert.

C. Android Would Have Existed in the But-For World

270. Google could have still provided Android by implementing the 37 API packages from the OpenJDK prior to Android's launch or by using C/C++ or another language as a programming language for Java (both of which Google actually has done). Thus, in the but-for world, Android would still have existed and would have captured a similar, if not the same, share of the smartphone market. As a result, the but-for world would have been essentially the

[427] Malackowski 1/8/2016 Report, ¶ 217.

[428] Expert Report of Dr. Cockburn, February 3, 2012, p. 255.

same as the actual world. Thus, Oracle's actual lost profits damages due to Google's alleged infringement are zero.

D. Adjustments to Mr. Malackowski's Calculation of Oracle's Lost Profits Damages

271. Mr. Malackowski calculates Oracle's lost profits damages relating to Java ME licensing due to Google's alleged infringement to be $475.4 million.[429] For all of the reasons discussed in the previous sections, Mr. Malackowski substantially overstates Oracle's lost profits for Java ME. In fact, for these same reasons, it is my opinion that Oracle's lost profits pertaining to Java ME licensing are zero. Furthermore, I note that Mr. Malackowski's claimed lost profits damages do not make economic sense on their face. In the initial years of the damages period—FY 2009 and FY 2010—sales of Android handsets were small. Yet, Mr. Malackowski claims there are significant lost profits damages. As a consequence, if one divides Mr. Malackowski's lost profits claim by the number of Android devices for each of FY 2009 and FY 2010, the result is absurdly high figures of $12.04 and $1.53 in damages per Android device, respectively. In contrast, in the following years, Mr. Malackowski's lost profits damages are in the range of $0.04 to $0.20 per Android device.[430] In the absence of any explanation for how the small number of Android handsets in the early years could have an effect on Java ME licensing that was an order of magnitude larger than effect of Android handsets in later years, Mr. Malackowski's claimed 2009 and 2010 lost profits are substantially overinflated.

272. Another flaw in Mr. Malackowski's calculation is that even though significant numbers of feature phones have continued to be sold over the years, Oracle has been unable to

[429] Malackowski 1/8/2016 Report,¶ 203.

[430] See Exhibit 4a.

generate the same amount of Java ME licensing revenue per feature phone handset. Java ME licensing revenue per feature phone handset has declined from $0.13 per handset in FY 2012 to just [redacted] in FY 2015.[431] Such decline in Java ME licensing revenue per feature phone could not have been the result of Android, or the alleged infringement, because Android is not competing with Java ME for use on feature phones. Instead, the decline in Java ME licensing revenue per feature phone handset is a result of Oracle being unable to license Java ME at a similar price and to as many feature phone OEMs as in the past, likely due to Sun's and later Oracle's unwillingness to invest in the Java ME platform and its resulting stagnation.

273. There is strong economic support for the conclusion that lost profits damages are zero. Nevertheless, to demonstrate the unreliability of Mr. Malackowski's calculation, I have made several adjustments described in the following paragraphs.

274. First, as previously discussed, Mr. Malackowski bases his calculation of lost profits damages on a comparison of Sun's forecasted Java ME licensing revenues to actual Java ME licensing revenues, and attributes the entire difference to the alleged infringement. Using these forecasts, which do not account for several reasons (as discussed throughout the previous sections) why Java ME licensing revenues and profits declined significantly and that have nothing to do with Android in general, or the alleged copyright infringement in particular, is inappropriate. Furthermore, Mr. Malackowski uses just one of the four Java ME licensing revenue forecast scenarios analyzed by Sun. For example, if Mr. Malackowski had used the Low or Medium forecast scenarios, and making no further adjustments to his approach to calculating Oracle's lost profits damages, then his lost profits damages would have been

[431] Exhibit 4b.

negative. Using the High forecast scenario, and making no further adjustments to Mr. Malackowski's approach to calculating Oracle's lost profits damages, results in lost profits damages much lower than $475.4 million.

275. Second, Mr. Malackowski also points to an October 2008 Sun document titled, Java in Wireless Business Review, to support his use of only the Strategic Forecast scenario from Sun's late 2007/early 2008 strategic forecasts. Specifically, Mr. Malackowski states:

> The "Best Estimate" forecast included in the Business Review projects Java ME revenue of $110.3 million in 2009 and $138.1 million in 2010, and is believed to be less encumbered by the existence of Android than the remaining scenarios which reflect lower projections. Use of the "Best Estimate" in this alternative forecast would result in much greater damages than I have calculated because the growth rate from 2009 to 2010 in the "Best Estimate" is greater than the 8% reflected in the Strategic Forecast.[432]

Contrary to Mr. Malackowski's claim, using the "Best Estimate" alternative forecast results in substantially lower damages than Mr. Malackowski calculated using Sun's Strategic Forecast. Mr. Malackowski states that damages would be "much greater" using the "Best Estimate" forecast because the 2009 to 2010 growth rate is greater than the 8% growth rate in the Strategic Forecast. But, there is no need to calculate and use a 2009 to 2010 growth rate when using the "Best Estimate" forecast to calculate lost profits damages because the actual forecast specifically projects Java ME licensing revenues from FY 2007 to FY 2012. Furthermore, although the 2009 to 2010 growth rate of 25% for Total ME Revenue in the "Best Estimate" forecast is greater than the 8% growth rate used by Mr. Malackowski for every year after 2010, the "Best Estimate" 2010 to 2011 growth rate declines to 11% while the 2011 to 2012 growth rate becomes negative at -6%. Focusing on just the CLDC Royalty and CDC – TV royalty portions

[432] Malackowski 1/8/2016 Report,¶ 187.

of Total ME Revenue in the "Best Estimate" forecast, which is consistent with the types of Java ME revenue that Mr. Malackowski uses to calculate actual Java ME revenue, the annual growth rates forecasted by Sun for 2010, 2011, and 2012 are 20%, 10%, and -12%, respectively.[433] Finally, focusing on just the CLDC Royalty portion of Total ME Revenue in the "Best Estimate" forecast, the annual growth rates forecasted by Sun from 2010 to 2012 are all negative. Thus, based on these "Best Estimate" forecasts, Sun was actually forecasting an eventual decline in Java ME licensing revenues.

276. Using the "Best Estimate" forecast scenario and the CLDC Royalty and CDC-TV royalty portions of Total ME Revenue from this forecast, and making no further adjustments to Mr. Malackowski's approach to calculating Oracle's lost profits damages, results in lost profits damages of $19.5 million.[434] Using the "Best Estimate" forecast scenario and only the CLDC Royalty portion of Total ME Revenue from this forecast, and making no further adjustments to Mr. Malackowski's approach to calculating Oracle's lost profits damages, results in negative damages.[435]

277. Third, I have adjusted Mr. Malackowski's calculation by starting with Oracle's actual Java ME licensing revenues, and asking what effect Android had on those revenues under conservative assumptions (including, for example, Mr. Malackowski's assumption that Android would not have existed in the but-for world, which is incorrect for the reasons discussed above). I have used two different approaches.

[433] OAGOOGLE0000142142 at 169.

[434] See Exhibit 4d.

[435] See Exhibit 4c.

278. The first approach is based on the following steps:

- I calculated the potential Java ME licensed handsets as a percentage of non-Android handsets, assuming that Android is not in the market. Potential Java ME licensed handsets is equal to all non-Android and non-iPhone handsets, and non-Android handsets is equal to potential Java ME licensed handsets plus iPhone handsets. For example, for FY 2009 this percentage equals 98.5%.[436]

- I calculate the number of Android units that would have been potential Java ME licensed handsets in the but-for world using this percentage. Then, I calculated the percentage increase in potential Java ME licensed handsets that these added handsets would have represented. For example, for FY 2009 this percentage was 0.2%.[437]

- Finally, I applied this percentage increase to the actual Java ME licensing revenues, and subtracted out incremental expenses, to calculate Java ME lost profits. For example, for FY 2009 this amount was $120,588. For the entire damages period from FY 2009 to FY 2015, total Java ME lost profits using this approach is equal to $128.5 million.[438]

279. The second approach is based on the following steps:

- I calculated a diversion ratio (as described above), which represents the percentage of Android units that would have become iPhone units in a but-for world without Android. This diversion ratio is equal to 36.6% on average. One minus the diversion ratio, or 63.4%, represents the percentage of Android units that would have become potential Java ME licensed handsets in the but-for world.

- I calculate the number of Android units that would have been potential Java ME licensed handsets in the but-for world using this percentage. Then, I calculated the percentage increase in potential Java ME licensed handsets that these added handsets would have represented. For example, for FY 2009 this percentage was 0.1%.

- Finally, I applied this percentage increase to the actual Java ME licensing revenues, and subtracted out incremental expenses, to calculate Java ME lost profits. For example, for FY 2009 this amount was $68,507. For the entire damages period from FY 2009 to FY 2015, total Java ME lost profits equal $85.7 million.[439]

[436] See Exhibit 4f.

[437] See Exhibit 4f.

[438] See Exhibit 4e.

[439] See Exhibit 4e.

280. I note that both of these approaches to calculating lost profits damages are conservative because they assume that in the but-for world that Android would not have existed, which is incorrect.

281. I also note that these figures are more consistent with both Sun's initial proposal to Google, and the parties' final negotiating positions, during the Sun-Google negotiations in 2005 and 2006 regarding a possible collaboration between the two companies to develop a Java-language-based mobile platform. Specifically, in February 2006, Sun proposed financial terms of $20 million per year for three years and 10% of Google revenue, with a cap of $25 million per year, to "cover the loss of revenue – one that we really cannot afford in the current economic condition, to fund the resources required to change the business where we are today to something unknown and untested."[440] Furthermore, Sun and Google eventually proposed a much smaller amount to cover, among other things, Sun's loss in Java ME licensing revenues, of only $28 million.[441] In fact, as Sun CEO Jonathan Schwartz testified, if necessary, Sun would have been willing to pay Google for the opportunity to be involved with Android, with the expectation that doing so would ultimately create downstream benefits for Sun's Java business.[442] Both Mr. Schwartz and other high-level Sun personnel expressed this expectation publicly at the time of Android's release.[443]

---

[440] Rubin 4/5/2011 Dep., Exhibit 8 (OAGOOGLE0000357494).

[441] See, e.g., OAGOOGLE0100166178-196 at 188; Rubin 7/27/2011 Dep., Exhibit PX316 (GOOGLE-12-00080356-367 at 359).

[442] Trial Tr. 1983:5-10 (Schwartz).

[443] "Congratulations Google, Red Hat and the Java Community!" Jonathan's Blog, November 5, 2007, http://web.archive.org/web/20101023072550/http://blogs.sun.com/jonathan/entry/congratulations_google; Barr Dep. Ex. 1366 (commenting that the release of Android shows that "Google recognized Java to be a

282. Importantly, any financial benefit Sun would have received would have been in exchange for the rights to use far more Sun intellectual property than just the SSO and declaring code for 37 API packages at issue here. The license under discussion between Sun and Google would have given Google rights to use the full Sun portfolio of hundreds of Java patents, all Java implementing software code (consisting of millions of lines of code), and the Java trademark. Moreover, although Sun contemplated that there would be a potential loss of Java ME licensing revenue under the Google-Sun collaboration, nothing in the history of the negotiations suggests that Sun linked any loss of Java ME licensing revenue to Google's use of the allegedly infringing works in this case. Rather, Sun perceived that such a loss might occur as a result of open sourcing the entire Java ME platform—which, as noted above, would involve making publicly available far more intellectual property than just the allegedly infringing works. Moreover, as discussed already, the accused copyrighted works in this case are the SSO and declaring code for 37 API packages from the Java SE platform, not anything from Java ME. Even before Google introduced Android, Sun had made the decision to open source the Java SE platform, and officially released an open source implementation of the vast majority of that platform, including the SSO and declaring code for the 37 API packages at issue here. To the extent Oracle is contending that Sun's Java ME licensing business was harmed by Google's use of a small fraction of the technology contained in the Java SE platform, that harm likely would have occurred anyway as a consequence of Sun's decision, before Android's release, to open source the entire Java SE platform, and cannot be attributed to Android.

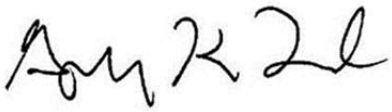
---

necessary part of [Android's] success" and that the use of Java in Android "literally guarantees Java remains the dominant platform for years to come").

VIII. Awarding Lost Profits in Addition to Unjust Enrichment Would Double-Count Damages

283. I understand that a plaintiff asserting copyright infringement may seek recovery of both the unjust enrichment to the alleged infringer and any actual damages suffered by the copyright owner, such as lost profits, as long as the latter are not duplicative of the former.

284. Here, the claimed actual damages are Oracle's lost profits arising from claimed lost Java ME licensing revenues. The claimed lost Java ME licensing revenues, in turn, arise from claimed displacement of Java ME by Android. Thus, the claimed lost Java ME profits are the flip side of the Android profits that Oracle claims as unjust enrichment. Accordingly, it would be duplicative to award both the claimed actual damages consisting of Java ME lost profits and the claimed unjust enrichment.

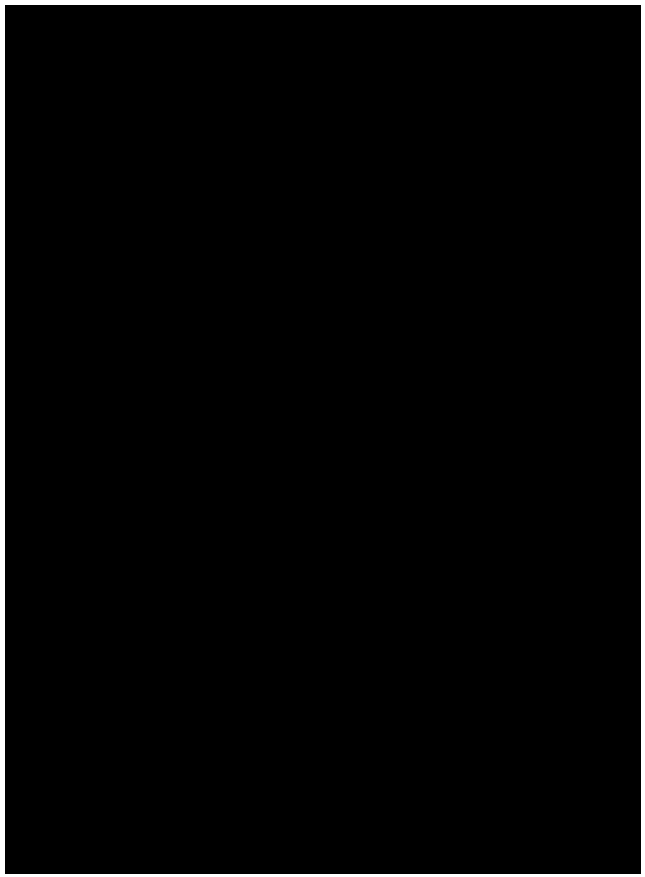


Gregory K. Leonard
March 10, 2016

Exhibits

Exhibit 1a.1
Android-Related Profits
Jan. 2008 - Dec. 2015

	2008 (a)	2009 (b)	2010 (c)	2011 (d)	2012 (e)	2013 (f)	2014 (g)	2015 (h)	Total (i)
Revenue									
Ads	$ 0.7	$ 15.7	$ 120.1	$ 569.4	$ 2,152.4				
Hardware	0.0	0.0	115.2	0.0	303.5				
Apps	0.0	1.1	8.0	36.2	136.1				
Digital Content	--	--	--	14.8	105.8				
Total	**$ 0.7**	**$ 16.8**	**$ 243.3**	**$ 620.4**	**$ 2,697.8**				
Cost of Sales									
TAC	$ 0.2	$ 2.9	$ 41.3	$ 108.3	$ 551.7				
Hardware	--	--	--	-0.2	340.6				
Apps	--	--	--	0.0	62.2				
Digital Content	--	--	--	23.5	169.5				
Infrastructure & Other COS	--	--	--	67.9	95.0				
Operations	0.2	0.5	4.3	--	--				
COS (including DTC)	0.0	0.3	109.9	--	--				
Total	**$ 0.4**	**$ 3.7**	**$ 155.5**	**$ 199.5**	**$ 1,219.0**				
Gross Profit									
Total Gross Profit	$ 0.3	$ 13.1	$ 87.9	$ 420.9	$ 1,478.8				
Gross Margin (%)	41.4 %	78.0 %	36.1 %	67.8 %	54.8 %				
Operating Expenses									
Android Engineering PM	$ 86.3	$ 43.1	$ 107.7	$ 192.3	$ 380.4				
Android Marketing	12.3	16.6	53.3	53.9	225.3				
Android Legal	1.0	2.1	32.2	160.5	113.7				
Android Sales and Other	0.9	3.2	5.2	16.3	37.3				
Android General and Administrative	--	26.8	42.8	126.0	124.7				
Incremental Search and Advertising Expenses	0.1	1.3	9.9	47.0	177.8				
Total	**$ 100.6**	**$ 93.1**	**$ 251.1**	**$ 596.1**	**$ 1,059.2**				
Profit	**$ (100.3)**	**$ (80.0)**	**$ (163.2)**	**$ (175.2)**	**$ 419.6**				



Notes: In millions of U.S. dollars.

Data from 2015 Q3 - 2015 Q4 is forecasted. Forecasts for all revenue and cost of sales and expense categories provided by Google, with the exception of Ads which is based on revenue from 2015 Q1 - Q2.

For TAC from 2011-2014 refer to Exhibit 1d. TAC for 2015 is based on 2014 percentage share.

For ads revenue for 2011 refer to GOOG-00132245 at 247 and GOOG-00132625 tab "2. Final - Legal."

Total Android legal expenses include a conservative deduction of [redacted] for legal expenses from the previous Google-Oracle litigation.

For Android general and administrative expenses refer to Exhibit 1e.

Incremental Search and Advertising expenses calculated from the product of Android Ads revenue and the coefficient for revenue of 0.08261 based on the regression of operating expenses and revenue for Search and Advertising Product Areas adjusted for the number of days per quarter and GNP inflation.

Sources: Exhibit 1d.

Exhibit 1e.

GOOG-00022386.

GOOGLE-00395614.

GOOG-00103813.

GOOG-00103815.

GOOG-00103812.

GOOG-00132625.

"Gross National Product: Implicit Price Deflator," U.S. Bureau of Economic Analysis, updated December 22, 2015.

"Android Update," August 20, 2012, GOOG-00132245.

Exhibit 1a.2
Android-Related Profits without Advertising
Jan. 2008 - Dec. 2015

	2008 (a)	2009 (b)	2010 (c)	2011 (d)	2012 (e)	2013 (f)	2014 (g)	2015 (h)	Total (i)
Revenue									
Hardware	$ 0.0	$ 0.0	$ 115.2	$ 0.0	$ 303.5				
Apps	0.0	1.1	8.0	36.2	136.1				
Digital Content	--	--	--	14.8	105.8				
Total	**$ 0.0**	**$ 1.1**	**$ 123.2**	**$ 51.0**	**$ 545.4**				
Cost of Sales									
Hardware	$ --	$ --	$ --	$ -0.2	$ 340.6				
Apps	--	--	--	0.0	62.2				
Digital Content	--	--	--	23.5	169.5				
Infrastructure & Other COS	--	--	--	67.9	95.0				
Operations	0.2	0.5	4.3	--	--				
COS (including DTC)	0.0	0.3	109.9	--	--				
Total	**$ 0.2**	**$ 0.8**	**$ 114.2**	**$ 91.2**	**$ 667.3**				
Gross Profit									
Total Gross Profit	$ (0.2)	$ 0.3	$ 9.0	$ (40.2)	$ (121.9)				
Gross Margin (%)	(985.3) %	24.7 %	7.3 %	(78.9) %	(22.4) %				
Operating Expenses									
Android Engineering PM	$ 86.3	$ 43.1	$ 107.7	$ 192.3	$ 380.4				
Android Marketing	12.3	16.6	53.3	53.9	225.3				
Android Legal	1.0	2.1	32.2	160.5	113.7				
Android Sales and Other	0.9	3.2	5.2	16.3	37.3				
Android General and Administrative	--	26.8	42.8	126.0	124.7				
Total	**$ 100.5**	**$ 91.8**	**$ 241.1**	**$ 549.0**	**$ 881.4**				
Profit	**$ (100.7)**	**$ (91.5)**	**$ (232.1)**	**$ (589.2)**	**$ (1,003.3)**				

Notes: In millions of U.S. dollars.
Data from 2015 Q3 - 2015 Q4 is forecasted. Forecasts for all revenue and cost of sales and expense categories provided by Google.
For Android general and administrative expenses refer to Exhibit 1e.
Total Android legal expenses include a conservative deduction of [redacted] for legal expenses from the previous Google-Oracle litigation.

Sources: Exhibit 1e.
GOOGLE-00395614.
GOOG-00103813.

Exhibit 1a.3
Android-Related Profits with iPhone Recapture Adjustment
Jan. 2008 - Dec. 2015

	2008 (a)	2009 (b)	2010 (c)	2011 (d)	2012 (e)	2013 (f)	2014 (g)	2015 (h)	Total (i)
Revenue									
Ads	$ 0.7	$ 15.7	$ 120.1	$ 569.4	$ 2,152.4				
Hardware	0.0	0.0	115.2	0.0	303.5				
Apps	0.0	1.1	8.0	36.2	136.1				
Digital Content	--	--	--	14.8	105.8				
Total	**$ 0.7**	**$ 16.8**	**$ 243.3**	**$ 620.4**	**$ 2,697.8**				
Cost of Sales									
TAC	$ 0.2	$ 2.9	$ 41.3	$ 108.3	$ 551.7				
Hardware	--	--	--	-0.2	340.6				
Apps	--	--	--	0.0	62.2				
Digital Content	--	--	--	23.5	169.5				
Infrastructure & Other COS	--	--	--	67.9	95.0				
Operations	0.2	0.5	4.3	--	--				
COS (including DTC)	0.0	0.3	109.9	--	--				
Total	**$ 0.4**	**$ 3.7**	**$ 155.5**	**$ 199.5**	**$ 1,219.0**				
Gross Profit									
Total Gross Profit	$ 0.3	$ 13.1	$ 87.9	$ 420.9	$ 1,478.8				
Gross Margin (%)	41.4 %	78.0 %	36.1 %	67.8 %	54.8 %				
Operating Expenses									
Android Engineering PM	$ 86.3	$ 43.1	$ 107.7	$ 192.3	$ 380.4				
Android Marketing	12.3	16.6	53.3	53.9	225.3				
Android Legal	1.0	2.1	32.2	160.5	113.7				
Android Sales and Other	0.9	3.2	5.2	16.3	37.3				
Android General and Administrative	--	26.8	42.8	126.0	124.7				
Incremental Search and Advertising Expenses	0.1	1.3	9.9	47.0	177.8				
Total	**$ 100.6**	**$ 93.1**	**$ 251.1**	**$ 596.1**	**$ 1,059.2**				
Profit Before iPhone Recapture Adjustment									
Total Operating Profit	$ (100.3)	$ (80.0)	$ (163.2)	$ (175.2)	$ 419.6				
iPhone Recapture Adjustment									
Profit									

Notes: In millions of U.S. dollars.

Data from 2015 Q3 - 2015 Q4 is forecasted. Forecasts for all revenue and cost of sales and expense categories provided by Google, with the exception of Ads which is based on revenue from 2015 Q1 - Q2.

For TAC from 2011-2014 refer to Exhibit 1d. TAC for 2015 is based on 2014 percentage share.

For ads revenue for 2011 refer to GOOG-00132245 at 247 and GOOG-00132625 tab "2. Final - Legal."

Total Android legal expenses include a conservative deduction of [redacted] for legal expenses from the previous Google-Oracle litigation.

For Android general and administrative expenses refer to Exhibit 1e.

Incremental Search and Advertising expenses calculated from the product of Android Ads revenue and the coefficient for revenue of 0.08261 based on the regression of operating expenses and revenue for Search and Advertising Product Areas adjusted for the number of days per quarter and GNP inflation.

For the iPhone Recapture Adjustment refer to Exhibit 1b.

Sources: Exhibit 1b.

Exhibit 1d.

Exhibit 1e.

GOOG-00022386.

GOOGLE-00395614.

GOOG-00103813.

GOOG-00103815.

GOOG-00103812.

GOOG-00132625.

"Gross National Product: Implicit Price Deflator," U.S. Bureau of Economic Analysis, updated December 22, 2015.

"Android Update," August 20, 2012, GOOG-00132245.

Exhibit 1a.4
Profit Apportioned to Android
Versus Search/Ads Technologies and Services

Revenue
- Android Search (w/ 32% Apportionment)
- Hardware
- Apps
- Digital Content

Total

Cost of Sales
- TAC
- Hardware
- Apps
- Digital Content
- Infrastructure & Other COS
- Operations
- COS (including DTC)

Total

Operating Expenses
- Android Engineering PM
- Android Marketing
- Android Legal
- Android Sales and Other
- Android General and Administrative
- Incremental Search and Advertising Expenses

Total

Android Profit

[redacted]

Notes: The 32% apportionment is calculated based on [redacted]
The annual margin impact figures of [redacted] and [redacted] are from GOOG-00130338 at 343.

Search revenue is from Exhibit 1c. Ads revenue for 2015 is based on revenue from 2015 Q1 - Q2.

Sources: Exhibit 1a.1.

Exhibit 1c.

GOOG-00130338.

Exhibit 1b
iPhone Recapture Adjustment
Jan. 2008 - Dec. 2015

	2008 (a)	2009 (b)	2010 (c)	2011 (d)	2012 (e)	2013 (f)	2014 (g)	2015 (h)	Total (i)
Revenue									
Ads	$ 0.7	$ 15.7	$ 120.1	$ 569.4	$ 2,152.4				
Hardware	--	--	--	--	--				
Apps	--	--	--	--	--				
Digital Content	--	--	--	--	--				
Total	**$ 0.7**	**$ 15.7**	**$ 120.1**	**$ 569.4**	**$ 2,152.4**				
Cost of Sales									
TAC	$ 0.2	$ 2.9	$ 41.3	$ 108.3	$ 551.7				
Hardware	--	--	--	--	--				
Apps	--	--	--	--	--				
Digital Content	--	--	--	--	--				
Infrastructure & Other COS	--	--	--	--	--				
Operations	--	--	--	--	--				
COS (including DTC)	--	--	--	--	--				
Total	**$ 0.2**	**$ 2.9**	**$ 41.3**	**$ 108.3**	**$ 551.7**				
Gross Profit									
Total Gross Profit	$ 0.5	$ 12.8	$ 78.9	$ 461.1	$ 1,600.7				
Gross Margin (%)	70.6 %	81.7 %	65.7 %	81.0 %	74.4 %				
Operating Expenses									
Android Engineering PM	$ --	$ --	$ --	$ --	$ --				
Android Marketing	--	--	--	--	--				
Android Legal	--	--	--	--	--				
Android Sales and Other	--	--	--	--	--				
Android General and Administrative	--	--	--	--	--				
Incremental Search and Advertising Expenses	0.1	1.3	9.9	47.0	177.8				
Total	**$ 0.1**	**$ 1.3**	**$ 9.9**	**$ 47.0**	**$ 177.8**				
Diversion Ratio	44.0 %	44.0 %	44.0 %	44.0 %	44.0 %				
Search Share	100.0	76.0	67.3	76.9	67.1				
iPhone Recapture Adjustment	**$ 0.1**	**$ 4.2**	**$ 25.7**	**$ 151.3**	**$ 532.5**				

Notes: In millions of U.S. dollars.

Ads revenue, cost of sales and incremental search and advertising expenses from Exhibit 1a.1.

For the diversion ratio refer to Exhibit 3d.2.

For the search share refer to search advertising as a share of Android advertising revenue from Exhibit 1c.

The iPhone recapture adjustment calculated as (diversion ratio)*[(Revenue-TAC)*(1-search share* [redacted])-incremental search and operating expenses].

The annual margin impact figures of [redacted] and [redacted] are from GOOG-00130338 at 343.

Sources: Exhibit 1a.1.

Exhibit 1c.

Exhibit 3d.2.

GOOG-00130338.

Exhibit 1c
Ads Revenue for 2011
Jan. 2008 - Jun. 2015

	2008 (a)	2009 (b)	2010 (c)	2011 (d)	2012 (e)	2013 (f)	2014 (g)	2015 Q1- Q2 (h)	Total (i)
Android Ads Revenue									
Search	$ 0.7	$ 11.9	$ 80.9	$ --	$ 1,444.9				
AdSense	0.0	0.0	6.8	--	238.6				
Display	0.0	3.8	32.4	--	468.9				
Total	**$ 0.7**	**$ 15.7**	**$ 120.1**	**$ --**	**$ 2,152.4**				
Android Ads Revenue (% of Total)									
Search	100.0 %	76.0 %	67.3 %	-- %	67.1 %				
AdSense	0.0	0.0	5.7	--	11.1				
Display	0.0	24.0	27.0	--	21.8				
Total	**100.0 %**	**100.0 %**	**100.0 %**	**-- %**	**100.0 %**				

[redacted]

2011 Ads Revenue Estimate Based on 2010-2012 Trend	Revenue	% of Total
Search	$ 437.9	76.9 %
AdSense	43.2	7.6
Display	88.3	15.5
Total	**$ 569.4**	**100.0 %**

Notes: In millions of U.S. dollars.

For Android Search ads revenue for 2011 refer to GOOG-00022388. Note that totals for Android Search Ads differ slightly between Google-00395614, GOOG-00022386 and GOOG-00022388.

For total revenue for 2011 refer to GOOG-00132245 at 247 and GOOG-00132625 tab "2. Final - Legal."

AdSense revenue for 2011 calculated based on the weighted average of (AdSense/(AdSense+Display)) for 2010 and 2012. The share is multiplied by (2011 Total Ads revenue - 2011 Search Ads revenue).

Display revenue for 2011 calculated based on the weighted average of (Display/(AdSense+Display)) for 2010 and 2012. The share is multiplied by (2011 Total Ads revenue - 2011 Search Ads revenue).

AFMA and AFMC is part of the Display category and AFMS is part of the AdSense category according to conversations with Jonathan Gold, Finance Director at Google.

Sources: Google-00395614.

GOOG-00022388.

GOOG-00022386.

GOOG-00132625.

"Android Update," August 20, 2012, GOOG-00132245.

Exhibit 1d
Traffic Acquisition Costs
Jan. 2011 - Dec. 2014

	2011 (a)	2012 (b)	2013 (c)	2014 (d)	Total (e)
Google Advertising TAC					
Adwords					
AFS					
Display					
Total					
Google Advertising Revenue					
Adwords					
AFS					
Display					
Total					
Google Advertising TAC (% of Revenue)					
Adwords					
AFS					
Display					
Android Advertising Revenue					
Search	$ 437.9	$ 1,444.9			
AdSense	43.2	238.6			
Display	88.3	468.9			
Total	**$ 569.4**	**$ 2,152.4**			
Android Advertising TAC					
Search	$ 23.3	$ 91.2			
AdSense	30.5	169.6			
Display	54.4	290.9			
Total	**$ 108.3**	**$ 551.7**			

Notes: In millions of U.S. dollars.

Based on conversations with Jonathan Gold, Finance Director at Google, Google AFS corresponds to the Android AdSense category, Google Adwords corresponds to the Android Search category, Google Display corresponds to the Android Display category.

Android advertising revenues by category for 2011 from Exhibit 1c.

Sources: Exhibit 1c.

GOOG-00022381.

GOOG-00022380.

GOOG-00022383.

GOOGLE-00395614.

GOOG-00022386.

Exhibit 1e
Android General and Administrative Expenses
Jan. 2009 - Dec. 2015

	2009 (a)	2010 (b)	2011 (c)	2012 (d)	2013 (e)	2014 (f)	2015 (g)	Total (h)
Google Headcount								
Android Engineers	[redacted]							
R&D Google Headcount								
Percentage Share								
Google General and Administrative Expenses								
Finance	[redacted]							
POps								
REWS								
Total								
Android General and Administrative Expenses								
Finance	[redacted]							
POps								
REWS								
Total								

Notes: General and Administrative expenses in millions of U.S. dollars.
Google G&A expenses for Business Operations and other G&A are not included.
R&D headcount increased 12% according to Google's 2015 3Q 10-Q.
Headcount for 2009 and 2010 measured at year-end.

Sources: GOOG-00103816.
GOOG-00103813.
GOOGLE-00395614.
Google Inc. 2015 3Q 10-Q, September 30, 2015, p. 39.
Google Inc. 2014 10-K, December 31, 2014, p. 5.
Google Inc. 2013 10-K, December 31, 2013, p. 7.
Google Inc. 2012 10-K, December 31, 2012, p. 9.
Google Inc. 2011 10-K, December 31, 2011, p. 8.
Google Inc. 2010 10-K, December 31, 2010, p. 8.
Google Inc. 2009 10-K, December 31, 2009, p. 18.
Google Inc. 2008 10-K, December 31, 2008, p. 17.

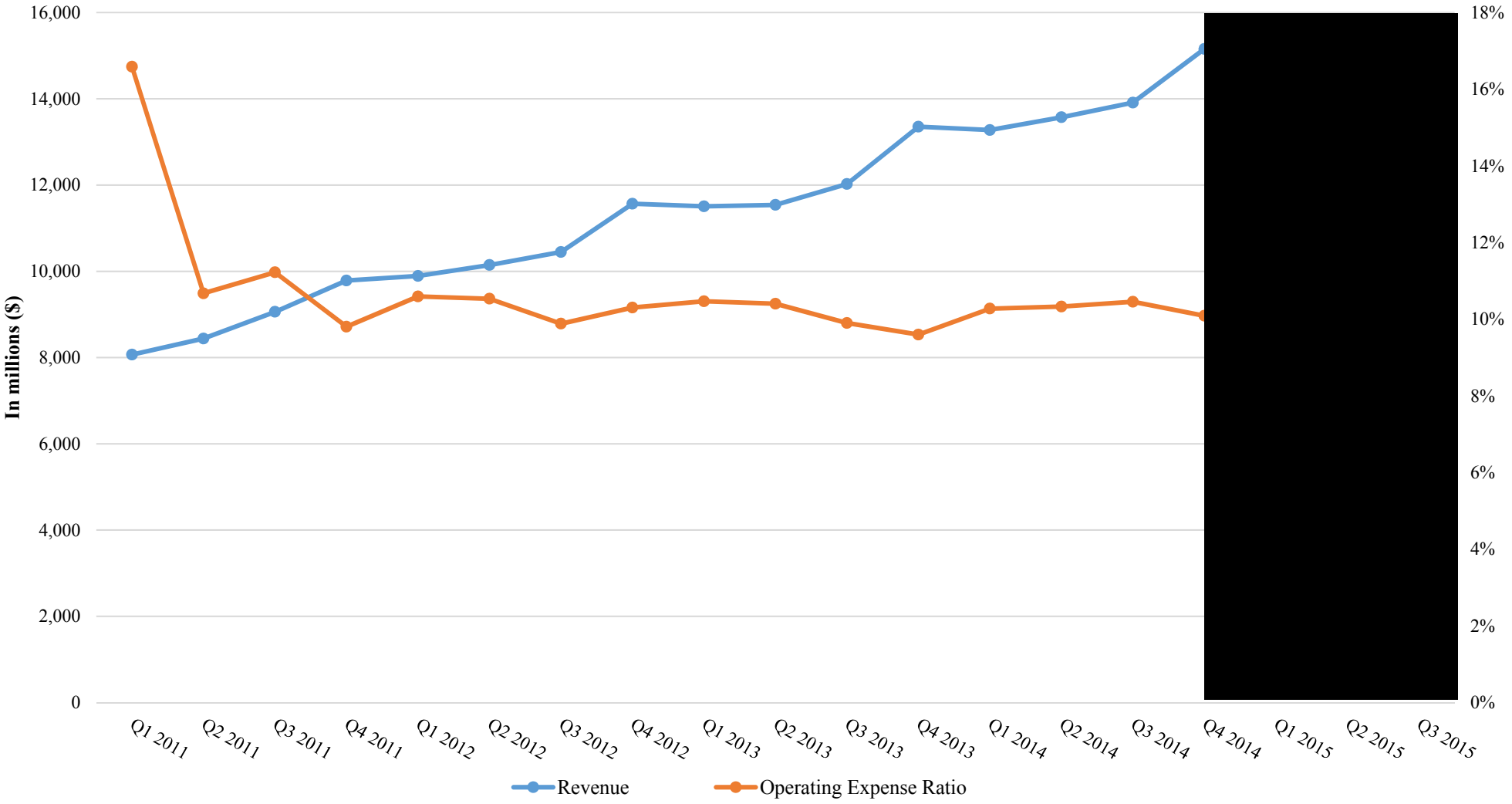
Exhibit 1f
Search and Advertising P&L
Jan. 2011 - Sep. 2015

	2011 (a)	2012 (b)	2013 (c)	2014 (d)	2015 Q1 - Q3 (e)	Total (f)
Revenue						
Search Product Area	[redacted]					
Advertising Product Area						
Total						
Cost of Sales						
TAC/CAC						
Data Center & Network						
Other Cost of Sales						
Total						
Gross Profit						
Total Gross Profit						
Gross Margin (%)						
Operating Expenses						
Engineering						
Sales						
Marketing						
Legal						
General and Administrative						
Total						
Operating Profit						
Total Operating Profit						
Operating Margin (%)						

Note: In millions of U.S. dollars.

Sources: GOOG-00103815.
GOOG-00103812.

Exhibit 1g
Search and Advertising Product Areas
Revenue and Operating Expense Ratio
Q1 2011 - Q3 2015



Sources: GOOG-00103815 and GOOG-00103812.

Exhibit 2a
Android Handset Price and Features

2010-2015

Quarter	Contract Price	Quality-Adjusted Contract Price	Number of Manufacturers	Number of Available Devices	Display Size (Inch)	Battery (mAh)	Camera Resolution (Megapixel)	Phone Height (Inch)	Phone Length (Inch)	Phone Width (Inch)	Memory (MB)	Share of Multi-Core Models
(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)	(l)	(m)
2010 Q1	$ 212.82	$ 212.82	3	15	3.35	1,387	4.58	0.56	4.51	2.28	277	0.00
2010 Q2	207.78	191.75	6	24	3.45	1,404	5.00	0.55	4.54	2.31	2,079	0.01
2010 Q3	228.94	177.76	7	37	3.71	1,454	5.30	0.50	4.67	2.40	7,047	0.00
2010 Q4	187.24	156.70	8	56	3.69	1,452	5.28	0.50	4.67	2.42	4,913	0.00
2011 Q1	154.67	133.19	8	67	3.72	1,457	5.21	0.50	4.67	2.43	4,136	0.02
2011 Q2	131.93	103.66	9	90	3.74	1,479	5.18	0.50	4.72	2.46	5,589	0.06
2011 Q3	129.91	91.81	13	118	3.77	1,501	5.44	0.49	4.73	2.47	6,592	0.22
2011 Q4	126.02	78.51	12	135	3.89	1,566	5.54	0.47	4.82	2.51	7,987	0.35
2012 Q1	123.11	66.94	12	116	3.90	1,624	5.63	0.45	4.85	2.53	9,055	0.39
2012 Q2	123.85	61.57	12	118	4.00	1,726	5.98	0.43	4.90	2.56	10,339	0.52
2012 Q3	118.82	61.41	14	136	4.15	1,804	6.29	0.42	4.99	2.60	11,068	0.63
2012 Q4	117.58	52.03	14	158	4.31	1,925	6.43	0.40	5.07	2.64	11,549	0.71
2013 Q1	129.83	52.81	14	153	4.31	1,970	6.28	0.40	5.04	2.64	11,210	0.73
2013 Q2	122.74	43.59	14	152	4.48	2,115	7.56	0.38	5.14	2.66	13,342	0.81
2013 Q3	120.95	42.73	12	164	4.49	2,158	7.73	0.38	5.13	2.66	13,039	0.83
2013 Q4	131.55	42.30	12	158	4.61	2,252	8.29	0.38	5.21	2.69	15,694	0.83
2014 Q1	125.07	41.22	11	135	4.61	2,245	8.18	0.38	5.21	2.69	15,189	0.84
2014 Q2	135.53	37.12	12	134	4.73	2,413	9.82	0.37	5.31	2.72	15,290	0.91
2014 Q3	118.20	33.59	13	135	4.73	2,428	9.72	0.37	5.31	2.71	15,079	0.94
2014 Q4	125.75	33.05	14	133	4.87	2,548	10.15	0.37	5.42	2.77	17,787	0.97
2015 Q1	118.80	32.71	13	117	4.90	2,545	9.50	0.37	5.45	2.79	17,067	0.98
2015 Q2	130.09	29.16	12	98	4.89	2,533	10.62	0.37	5.46	2.77	35,227	1.00
2015 Q3	101.32	17.68	12	103	4.94	2,517	10.52	0.36	5.52	2.78	32,530	1.00
2015 Q4	87.81	13.58	13	102	5.05	2,564	10.88	0.34	5.62	2.82	36,013	1.00

Notes: Contract price and features of individual handsets are summed, weighting by quarterly sales of Android smartphones.
Quality adjusted prices are calculated using a chained Fisher price indices.

Source: "ITG Monthly Mobile Handset Report," ITG, December 2015.

Exhibit 2b
Programming Languages
of Multi-Homed Applications Appearing on Monthly Top 100 Download Lists

2012-2013

No.	Android Applications	Programming Language
(a)	(b)	(c)
1	4 Pics 1 Song	NDK
2	Angry Birds	NDK
3	Angry Birds Friends	NDK
4	Angry Birds Go!	NDK
5	Angry Birds Rio	NDK
6	Angry Birds Seasons	NDK
7	Angry Birds Space	NDK
8	Angry Birds Star Wars	NDK
9	Angry Birds Star Wars II	NDK
10	Arcane Empires	NDK
11	Asphalt 7: Heat	NDK
12	Asphalt 8: Airborne	NDK
13	Avengers Alliance	NDK
14	Backflip Madness	NDK
15	Bad Piggies	NDK
16	BBM	NDK
17	Beard Salon - Free games	NDK
18	Beat the Boss 2	NDK
19	Bejeweled Blitz	NDK
20	Bitstrips	NDK
21	Blood Brothers (RPG)	NDK
22	Bloons TD 5	NDK
23	Bubble Witch Saga	NDK
24	Can You Escape	NDK
25	Candy Crush Saga	NDK
26	Castle Clash: Age of Legends	NDK
27	CastleVille Legends	NDK
28	Clash of Clans	NDK
29	Clear Vision (17+)	NDK
30	Clear Vision 2	NDK
31	Crazy Dentist - Fun games	NDK
32	CSR Racing	NDK
33	Cut the Rope: Experiments	NDK
34	Dark Summoner	NDK
35	DEAD TRIGGER	NDK
36	Diamond Dash	NDK

No.	Android Applications	Programming Language
(a)	(b)	(c)
37	Diptic	NDK
38	Dots: A Game About Connecting	NDK
39	Drag Racing Classic	NDK
40	DragonVale	NDK
41	Draw Something	NDK
42	Draw Something Free	NDK
43	Dropbox	NDK
44	Dumb Ways to Die	NDK
45	Earn to Die	NDK
46	eBay	NDK
47	ESPN	NDK
48	Farming Simulator 14	NDK
49	Fashion Story™	NDK
50	Flow Free	NDK
51	Frozen Free Fall	NDK
52	Fruit Ninja	NDK
53	Fruit Ninja Free	NDK
54	Fun Run - Multiplayer Race	NDK
55	Gangstar Rio: City of Saints	NDK
56	Google Earth	NDK
57	Google Translate	NDK
58	Grand Theft Auto III	NDK
59	Grand Theft Auto: San Andreas	NDK
60	Grand Theft Auto: Vice City	NDK
61	Guncrafter	NDK
62	Hard Time (Prison Sim)	NDK
63	Hardest Game Ever 2	NDK
64	Hay Day	NDK
65	Hill Climb Racing	NDK
66	Hungry Shark Evolution	NDK
67	Ice Age Village	NDK
68	Injustice: Gods Among Us	NDK
69	Instagram	NDK
70	Into the Dead	NDK
71	Iron Man 3 - The Official Game	NDK
72	Jackpot Party Casino - Slots	NDK
73	Jelly Splash	NDK
74	Jetpack Joyride	NDK
75	Juice Cubes	NDK
76	Jurassic Park™ Builder	NDK
77	Kik	NDK
78	Kingdom Rush Frontiers	NDK
79	Lazors	NDK

No.	Android Applications	Programming Language
(a)	(b)	(c)
80	Legend of the Cryptids	NDK
81	Line Runner	NDK
82	LinkedIn	NDK
83	Magic Piano by Smule	NDK
84	MARVEL War of Heroes	NDK
85	Max Payne Mobile	NDK
86	Mega Jump	NDK
87	Megapolis	NDK
88	Messenger	NDK
89	Minecraft: Pocket Edition	NDK
90	MLB.com At Bat	NDK
91	Modern Combat 3: Fallen Nation	NDK
92	Modern Combat 4: Zero Hour	NDK
93	Monster Doctor - kids games	NDK
94	Moto X Mayhem	NDK
95	My Singing Monsters	NDK
96	My Talking Tom	NDK
97	NBA 2015-16	NDK
98	NCAA March Madness Live	NDK
99	Netflix	NDK
100	Order & Chaos Online	NDK
101	Papa Pear Saga	NDK
102	Parking Frenzy 2.0	NDK
103	Path	NDK
104	Pet Rescue Saga	NDK
105	Photo Editor by Aviary	NDK
106	Pinterest	NDK
107	Plague Inc.	NDK
108	Plants vs. Zombies™ 2	NDK
109	Rayman Jungle Run	NDK
110	Real Basketball	NDK
111	Real Boxing	NDK
112	Real Racing 3	NDK
113	Real Steel	NDK
114	Riptide GP2	NDK
115	Robot Unicorn Attack 2	NDK
116	Ruzzle	NDK
117	Samurai Siege: Alliance Wars	NDK
118	Scribblenauts Remix	NDK
119	Shark Dash	NDK
120	SimpleRockets	NDK
121	Ski Safari	NDK
122	Slots Journey	NDK

No.	Android Applications	Programming Language
(a)	(b)	(c)
123	Snapchat	NDK
124	SongPop	NDK
125	Sonic Dash	NDK
126	Sonic The Hedgehog	NDK
127	SoundCloud - Music & Audio	NDK
128	Stickman Cliff Diving	NDK
129	Subway Surfers	NDK
130	Survivalcraft	NDK
131	Talking Tom Cat 2	NDK
132	Temple Run	NDK
133	Temple Run 2	NDK
134	Temple Run: Brave	NDK
135	Temple Run: Oz	NDK
136	The Amazing Spider-Man	NDK
137	The Impossible Game	NDK
138	The Lost City	NDK
139	The Simpsons™: Tapped Out	NDK
140	The Sims 3	NDK
141	The Sims™ FreePlay	NDK
142	Tiny Village	NDK
143	Toca Hair Salon 2	NDK
144	Top Girl	NDK
145	Trial Xtreme 3	NDK
146	Tumblr	NDK
147	Turbo FAST	NDK
148	Twitter	NDK
149	Unblock Me FREE	NDK
150	Viber	NDK
151	Voxer Walkie Talkie Messenger	NDK
152	WatchESPN	NDK
153	WhatsApp Messenger	NDK
154	Wheres My Mickey? Free	NDK
155	Wheres My Water? 2	NDK
156	Wild Blood	NDK
157	Wipeout	NDK
158	WolframAlpha	NDK
159	Word Search Puzzles	NDK
160	World War™	NDK
161	YouTube	NDK
162	Zombie Highway	NDK
163	Bible	Java
164	Bubble Mania™	Java
165	Comics	Java
166	DC Comics	Java

No.	Android Applications	Programming Language
(a)	(b)	(c)
167	DOOORS - room escape game -	Java
168	Dungeon Village	Java
169	ESPN Fantasy Football	Java
170	Geocaching	Java
171	Hanging With Friends	Java
172	Heads Up!	Java
173	HotSchedules	Java
174	IRS2Go	Java
175	Lord of the Dragons	Java
176	Marvel Comics	Java
177	Move the Box	Java
178	OkCupid Dating	Java
179	PicFrame	Java
180	Robot Unicorn Attack	Java
181	SimSimi	Java
182	Solitaire	Java
183	The Oregon Trail	Java
184	Word Streak With Friends	Java
185	Yelp	Java
186	Doodle Jump	Java
187	Facebook	Java
188	The Moron Test	Java
189	4 Pics 1 Word	Middleware
190	Icomania	Middleware
191	Talking Pierre the Parrot	Middleware
192	Toms Love Letters	Middleware
193	ZEDGE™ Ringtones & Wallpapers	Middleware
194	Amazing Alex	Unknown
195	Asphalt 6: Adrenaline	Unknown
196	Assassins Creed Pirates	Unknown
197	BIG WIN Football	Unknown
198	Bloons TD 4	Unknown
199	CONTRA： EVOLUTION	Unknown
200	D.O.T. Defender of Texel (RPG)	Unknown
201	Devils Attorney	Unknown
202	Dragon Story: New Dawn	Unknown
203	Dragon Story: Valentines Day	Unknown
204	Dragon Story™	Unknown
205	Dream Heights	Unknown
206	Dream Zoo	Unknown
207	Family Feud® & Friends	Unknown
208	Fast & Furious 6: The Game	Unknown
209	Jaws™ Revenge	Unknown
210	MADDEN NFL 12 by EA SPORTS™	Unknown
211	MLB Big Stars Baseball	Unknown
212	MONOPOLY Millionaire	Unknown
213	Monsters University	Unknown
214	NFL Flick Quarterback	Unknown
215	NFL Kicker 13	Unknown
216	NFL Kicker!	Unknown
217	Pet Shop Story™	Unknown

No.	Android Applications	Programming Language
(a)	(b)	(c)
218	Rage of Bahamut	Unknown
219	Slots by Zynga	Unknown
220	Smurfs Village	Unknown
221	SongPop Plus	Unknown
222	Star Wars: Tiny Death Star	Unknown
223	Tap Zoo	Unknown
224	Toy Story: Smash It!	Unknown
225	Ugly Meter	Unknown
226	Whats My IQ?™	Unknown
227	Wheres My Mickey?	Unknown
228	Wheres My Perry? Free	Unknown
229	Wheres My Water?	Unknown
230	Wheres My Water? Free	Unknown
231	World Series of Poker	Unknown
232	Wreck-it Ralph	Unknown
233	ZENONIA® 4	Unknown
234	Zombies, Run!	Unknown

Notes: There are 162 NDK apps.
There are 26 Java apps.
There are 5 Middleware apps.
There are 41 unknown apps.

Sources: "Market Data,"App Annie, data as of December 18, 2015.
Programming languages are identified by Google.
Expert Report of Dr. Cox, Exhibit 1b and Exhibit 1c, October 3, 2011.

Exhibit 2c
Analysis of The Lag in Application Launch Date Between Google Play and Apple Store
2012-2013

	Base Model (a)	Base Model with Time Trend (b)	Base Model with Time Trend and Interaction of Java with Android Share (c)	Base Model with App Type and App Category Dummies (d)	Base Model with App Type and App Category Dummies and Time Trend (e)	Base Model with App Type and App Category Dummies and Time Trend and Interaction of Java with Android Share (f)
Coefficient						
Java	255.732***	-33.009	104.403	138.781*	-12.917	50.984
	(72.548)	(44.364)	(74.198)	(73.899)	(50.365)	(77.925)
Android Relative Share to iOS	-209.738***	-93.321***	-73.051***	-147.778***	-81.688***	-75.136***
	(43.762)	(20.409)	(19.796)	(44.667)	(20.599)	(20.728)
Time Trend		-17.862***	-17.555***		-16.841***	-16.767***
		(0.718)	(0.766)		(1.156)	(1.183)
Jave*Android Relative Share to iOS			-97.421*			-43.727
			(52.615)			(61.401)
Paid Apps Indicator				11.181	-44.267	-44.776
				(45.709)	(29.582)	(29.781)
Product Category Indicators						
Books & Reference				171.821	53.416	33.565
				(125.021)	(71.599)	(57.535)
Brain & Puzzle				-685.507***	-162.929***	-158.329***
				(72.846)	(42.600)	(41.999)
Casual				74.832	90.114	54.901
				(118.003)	(56.132)	(58.611)
Comics				-305.514***	-216.790***	-224.752***
				(105.916)	(71.861)	(78.385)
Education				-536.761***	33.848	39.965
				(91.640)	(52.646)	(52.549)
Entertainment				-399.175***	-157.557**	-154.364**
				(125.609)	(66.365)	(66.071)
Finance				-559.957***	-389.294***	-336.821***
				(107.678)	(66.461)	(115.437)
Game				-487.563***	-85.043*	-81.615*
				(73.308)	(43.991)	(43.822)
Health & Fitness				208.467*	140.603**	105.163*
				(118.393)	(56.786)	(59.005)
Media & Video				-954.477***	-645.477***	-639.389***
				(74.215)	(32.448)	(31.538)
Music & Audio				10.598	10.754	14.115
				(122.331)	(28.639)	(29.423)
Personalization				-973.725***	-662.979***	-658.461***
				(70.763)	(32.907)	(32.246)
Photography				-436.225**	-208.694*	-194.360
				(169.909)	(124.117)	(132.129)

	Base Model (a)	Base Model with Time Trend (b)	Base Model with Time Trend and Interaction of Java with Android Share (c)	Base Model with App Type and App Category Dummies (d)	Base Model with App Type and App Category Dummies and Time Trend (e)	Base Model with App Type and App Category Dummies and Time Trend and Interaction of Java with Android Share (f)
Productivity				106.168	30.164	19.534
				(125.038)	(90.534)	(74.195)
Shopping				388.736***	83.638	100.317*
				(105.979)	(51.645)	(53.181)
Sports				-250.745	-169.647*	-153.381*
				(170.415)	(94.173)	(79.040)
Tools				-283.337***	-174.562***	-170.892***
				(68.528)	(27.584)	(27.131)
Travel & Local				326.692***	112.487**	96.092**
				(99.332)	(47.696)	(43.058)
Constant	533.365***	1,140.536***	1,095.437***	855.264***	1,177.203***	1,160.450***
	(77.392)	(45.622)	(51.401)	(105.979)	(52.387)	(53.991)
Observations	193	193	193	193	193	193
R-squared	0.246	0.779	0.785	0.533	0.829	0.830

Notes: The sample includes monthly top 100 free/paid downloaded apps in 2012 and 2013, that were launched on both iOS and Android platforms. Apps with programming language unknown are excluded from the regression analysis.

The dependent variable is the difference in days between the launch date on Google Play and on Apple Store.

Time trend is a linear trend based on the earlier launch date.

Omitted app categories are Communications and Social in columns (d) - (f).

Robust standard errors are in parentheses.

Programming languages are identified based on information provided by Google, and Android application languages in the Export Report of Dr. Cox.

Sources: "Market Data,"App Annie, data as of December 18, 2015.

Expert Report of Dr. Cox, Exhibit 1b and Exhibit 1c, October 3, 2011.

Expert Report of Dr. Leonard, February 8, 2016.

Exhibit 2d
Time Spent on Android Apps

	Jan. 2013 (a)	Feb. 2013 (b)	Mar. 2013 (c)	Jan. 2015 (d)	Feb. 2015 (e)	Mar. 2015 (f)
Percentage of Total Minutes Spent on						
Top 5 Apps	40.4 %	34.6 %	36.7 %	37.5 %	38.3 %	40.4 %
Top 10 Apps	51.1	47.2	51.0	45.1	46.6	48.2
Top 50 Apps	74.7	72.7	78.1	68.0	69.6	70.4
Top 100 Apps	81.1	79.9	85.8	76.7	78.0	78.8
Top 200 Apps	86.4	86.2	92.5	84.4	85.5	86.2

Note: The data includes a total of 791 Android apps in 2013, and 1,464 Android apps in 2015.

Source: "Mobile Metrix," comScore, Inc., January 19, 2016.

Exhibit 2e
Top Developers Ranked by Free App Downloads

2012-2015

2012			2013			2014			2015		
Developers	**Downloads**	**Dual-Homed**	**Developers**	**Downloads**	**Dual-Homed**	**Developers**	**Downloads**	**Dual-Homed**	**Developers**	**Downloads**	**Dual-Homed**
Google	200,619,752	Yes	Google	112,458,399	Yes	Facebook	125,585,192	Yes	Google	131,797,294	Yes
GO Dev Team	78,232,673	No	Facebook	103,931,618	Yes	King	91,769,172	Yes	Facebook	106,749,382	Yes
Rovio	55,907,805	Yes	GO Dev Team	59,460,049	No	Google	88,363,443	Yes	King	62,948,134	Yes
Facebook	54,890,622	Yes	Rovio	54,734,033	Yes	Pandora	46,033,945	Yes	Cheetah Mobile	47,586,663	Yes
Adobe	31,675,168	Yes	King	54,495,732	Yes	6677g	42,946,944	Yes	Gameloft	42,645,488	Yes
Zynga	29,539,304	Yes	Pandora	45,328,953	Yes	Gameloft	42,792,220	Yes	Outfit7	42,554,397	Yes
Pandora	25,134,437	Yes	Electronic Arts	44,046,306	Yes	Rovio	42,264,513	Yes	Firemint	40,449,584	Yes
DroidHen	24,355,233	Yes	6677g	43,192,947	Yes	Electronic Arts	42,100,525	Yes	Instagram	37,328,997	Yes
Amazon	22,293,828	Yes	Instagram	38,330,884	Yes	Outfit7	41,524,720	Yes	Glu	34,690,252	Yes
Yahoo!	21,211,950	Yes	Gameloft	34,584,194	Yes	Cheetah Mobile	40,906,154	Yes	Microsoft	34,665,904	Yes
MagmaMobile.com	20,732,660	Yes	Imangi	32,353,059	Yes	Glu	39,377,442	Yes	Pandora	33,760,642	Yes
Outfit7	19,248,292	Yes	Amazon	31,895,636	Yes	Instagram	38,513,399	Yes	Amazon	33,663,687	Yes
Runner Games	19,001,287	No	Team Lava	30,098,172	Yes	IGG	37,332,115	Yes	Rovio	33,090,300	Yes
Gameloft	18,926,920	Yes	Yahoo!	28,103,505	Yes	GO Dev Team	31,700,883	No	Miniclip	32,984,318	Yes
Skype	18,070,424	Yes	Skype	27,728,713	Yes	Snapchat	29,631,857	Yes	Disney	32,769,758	Yes
Freedom Design	17,767,069	No	Zynga	27,052,266	Yes	Supercell	28,768,610	Yes	6677g	32,220,550	Yes
Twitter	16,325,359	Yes	Netflix	25,513,086	Yes	Yahoo!	28,603,008	Yes	Snapchat	28,884,898	Yes
Instagram	16,321,587	Yes	Kiloo	24,579,166	Yes	Netflix	28,418,206	Yes	GO Dev Team	27,923,200	No
Halfbrick	15,078,159	Yes	Twitter	24,092,849	Yes	Surpax	28,007,609	Yes	Supercell	25,551,663	Yes
Glu	15,051,614	Yes	Adobe	24,018,626	Yes	Disney	27,710,528	Yes	IGG	25,395,298	Yes
Netflix	15,011,635	Yes	Outfit7	23,330,804	Yes	Zynga	26,344,829	Yes	Zynga	25,347,904	Yes
Words Mobile	14,882,263	No	Halfbrick	22,875,680	Yes	Amazon	25,922,338	Yes	Surpax	24,881,972	Yes
Zedge	14,654,385	Yes	Disney	22,689,502	Yes	Skype	24,728,174	Yes	Ketchapp Studio	23,165,643	Yes
Team Lava	14,438,416	Yes	DroidHen	21,246,738	Yes	Kik	24,138,447	Yes	ContextLogic	23,162,148	Yes
AppTornado	12,233,550	No	Kik	21,062,538	Yes	Imangi	21,677,631	Yes	Netflix	23,062,978	Yes
Imangi	12,085,936	Yes	Snapchat	20,478,153	Yes	Kiloo	20,760,261	Yes	Spotify	21,323,759	Yes
ESPN	12,064,905	Yes	Game Circus	19,113,327	Yes	WhatsApp	19,919,834	Yes	TabTale	21,303,766	Yes
Electronic Arts	11,998,457	Yes	eBay	18,789,199	Yes	Twitter	19,162,500	Yes	SGN	20,713,297	Yes
Creative Mobile	11,516,075	Yes	ESPN	18,380,572	Yes	Spotify	18,063,395	Yes	Yahoo!	20,435,985	Yes
Voxer	10,986,545	Yes	TangoMe	17,534,624	Yes	Halfbrick	17,921,563	Yes	WhatsApp	20,128,990	Yes
TangoMe	10,974,930	Yes	Surpax	17,285,003	Yes	Miniclip	17,906,905	Yes	Qihoo 360	19,815,096	No
TheWeatherChannel	10,889,687	Yes	Zedge	17,140,777	Yes	ESPN	17,476,858	Yes	Kik	19,762,277	Yes
iTreeGamer	10,829,757	No	FT Games	16,672,694	Yes	TabTale	17,361,359	Yes	Imangi	19,114,684	Yes
Nikolay Ananiev	10,473,631	Yes	Fingersoft	16,335,546	Yes	Team Lava	17,288,469	Yes	Kiloo	19,020,328	Yes
Disney	10,288,987	Yes	Big Duck	15,462,216	Yes	eBay	17,195,453	Yes	Zentertain	18,165,967	Yes
GAMEVIL	10,267,947	Yes	TheWeatherChannel	15,341,107	Yes	Adobe	16,667,393	Yes	Skype	17,993,083	Yes
GoldenShores	10,197,691	No	Clear Channel	14,414,826	No	Zedge	15,615,888	Yes	Etermax	17,833,003	Yes
Lookout	10,064,865	Yes	Glu	14,244,330	Yes	dotGears Studios	15,472,221	Yes	Storm8	16,830,543	Yes
MobilityWare	9,958,556	Yes	WhatsApp	13,971,265	Yes	TangoMe	15,324,533	Yes	Yodo1	16,669,378	Yes
eBay	9,898,039	Yes	MobilityWare	13,296,336	Yes	LINE	14,635,244	Yes	Electronic Arts	15,747,481	Yes
Shazam Entertainment	9,556,469	Yes	LifeChurch.tv	13,264,245	Yes	Fingersoft	14,382,294	Yes	Pinterest	15,725,113	Yes
TerranDroid	9,505,294	No	Shazam Entertainment	13,197,062	Yes	Pinterest	14,096,298	Yes	Game Circus	15,493,152	Yes
Com2uS	9,480,747	Yes	Pinterest	12,971,612	Yes	MobilityWare	13,140,446	Yes	Halfbrick	15,426,858	Yes
PicsArt	9,401,742	Yes	PicsArt	12,753,672	Yes	Zentertain	13,065,020	Yes	Warner Bros	15,015,610	Yes
ReChild	9,353,644	No	RoidApp	12,452,736	Yes	Mobage	12,648,143	Yes	Zedge	14,776,444	Yes
Fingersoft	9,141,802	Yes	IGG	12,150,924	Yes	VascoGames	12,081,170	No	Ludia	14,410,325	Yes
WhatsApp	9,112,237	Yes	ZeptoLab	12,143,960	Yes	Smule	11,918,094	Yes	Twitter	14,313,667	Yes
Handcent	9,109,049	No	GoldenShores	11,829,138	No	Microsoft	11,700,954	Yes	Smule	13,802,112	Yes
ZXing Team	9,055,887	No	Creative Mobile	11,391,880	Yes	LEGO	11,609,821	Yes	VascoGames	13,671,073	No
Game Circus	9,005,294	Yes	TuneIn	11,356,717	Yes	Kakapo Games	11,572,018	Yes	SoundCloud	12,991,867	Yes
OMGPOP	8,966,238	Yes	Mobage	11,324,346	Yes	TheWeatherChannel	11,012,996	Yes	DU Apps	12,632,972	No
Kittehface	8,933,537	Yes	GREE	11,314,216	Yes	GREE	10,656,020	Yes	Adobe	12,339,457	Yes
Melimots	8,818,997	No	ooVoo	10,469,750	Yes	SoundCloud	10,650,622	Yes	Fingersoft	11,536,336	Yes
2Easy Team	8,654,305	No	YYH Creative	10,407,826	Yes	Libii	10,563,577	Yes	TheWeatherChannel	10,677,346	Yes
6677g	8,486,511	Yes	Words Mobile	10,396,226	No	Shazam Entertainment	9,987,962	Yes	Tim O	10,627,569	Yes
Backflip Studios	8,422,871	Yes	Lookout	10,370,553	Yes	Playtika	9,749,032	Yes	iHeartMedia	10,389,372	Yes
City Games	8,291,811	No	Nikolay Ananiev	10,263,384	Yes	ZeptoLab	9,736,560	Yes	ESPN	9,901,809	Yes
Italy Games	8,277,190	No	Social and Mobile	10,206,870	No	Pinger	9,677,067	Yes	General Adaptive Apps	9,771,411	Yes
AI Factory	8,206,913	No	Italy Games	10,100,244	No	GAMEVIL	9,600,208	Yes	LEGO System A/S	9,715,578	Yes
FT Games	7,956,052	Yes	LOTUM	10,085,372	Yes	Random Logic	9,582,019	Yes	Metro Trains	9,438,012	Yes
Mobage	7,823,467	Yes	ReChild	9,817,052	No	Vine Labs	9,466,636	Yes	Libii	9,363,838	Yes
Bubble Quiz	7,820,275	Yes	Ludia	9,805,351	Yes	SGN	9,358,803	Yes	Tapinator	9,241,780	Yes
Clear Channel	7,752,553	No	Miniclip	9,786,660	Yes	iHeartMedia	9,266,481	Yes	Scottgames	9,200,094	Yes
Social and Mobile	7,715,567	No	MagmaMobile.com	9,712,761	Yes	Game Circus	9,234,611	Yes	Nordcurrent	9,157,523	Yes
LifeChurch.tv	7,613,211	Yes	KS Mobile	9,553,783	Yes	Big Duck	9,046,540	Yes	Mobile Motion	9,149,304	Yes
TuneIn	7,249,189	Yes	Vine Labs	9,433,653	Yes	Italy Games	8,831,648	No	Big Fish Games	9,034,402	Yes
Slacker	6,773,988	Yes	Ace Viral	9,048,965	Yes	Ludia	8,794,704	Yes	LINE	8,928,006	Yes
Big Duck	6,103,186	Yes	GAMEVIL	8,927,849	Yes	KS Mobile	8,576,220	Yes	Super Lucky Casino	8,848,385	Yes
Ace Viral	6,094,981	Yes	TabTale	8,789,325	Yes	SEGA	8,420,925	Yes	LEGO	8,706,482	Yes
Natenai Ariyatrakool	5,542,449	Yes	Groupon	8,578,548	Yes	LifeChurch.tv	8,288,591	Yes	Hothead Games	8,635,458	Yes
Miniclip	5,155,691	Yes	Spotify	8,556,312	Yes	RoidApp	8,242,612	Yes	Uber Technologies	8,506,902	Yes

2012			2013			2014			2015		
Developers	**Downloads**	**Dual-Homed**	**Developers**	**Downloads**	**Dual-Homed**	**Developers**	**Downloads**	**Dual-Homed**	**Developers**	**Downloads**	**Dual-Homed**
Djinnworks	5,069,652	Yes	Pinger	8,408,808	Yes	Djinnworks	8,209,430	Yes	MobilityWare	8,479,247	Yes
J2 Interactive	4,942,654	No	Freedom Design	8,168,826	No	Firemint	8,083,933	Yes	Pinger	8,183,010	Yes
LuckyStar	4,864,832	No	Dropbox	7,497,250	Yes	ooVoo	7,753,439	Yes	Machine Zone	8,140,536	Yes
Samsung Electronics	4,799,901	Yes	MAG Interactive	7,304,798	Yes	Warner Bros	7,453,644	Yes	NBC Universal	7,941,488	Yes
Sunfoer	4,645,309	No	Handcent	6,893,408	No	Metro Trains	7,252,698	Yes	SEGA	7,685,153	Yes
K-Factor	4,583,955	Yes	Aviary	6,592,615	Yes	Kabam	7,123,753	Yes	Samsung Electronics	7,477,226	Yes
BestCoolFunGames	4,283,330	Yes	iTreeGamer	6,339,239	No	AVG	7,055,280	No	Scopely	7,320,678	Yes
Pinterest	4,281,678	Yes	J2 Interactive	6,337,178	No	Com2uS	6,994,754	Yes	Budge	7,146,780	No
Progimax	4,124,444	No	NFL Enterprises	6,265,741	Yes	Ketchapp Studio	6,694,059	Yes	TangoMe	7,044,420	Yes
ZeptoLab	4,000,385	Yes	Wooga	6,232,392	Yes	PicsArt	6,678,026	Yes	Djinnworks	7,019,041	Yes
Opera Software	3,889,430	Yes	Libii	6,154,527	Yes	Scopely	6,398,556	Yes	Playtika	6,975,689	Yes
Flixster	3,827,087	Yes	Big Bang INC.	6,061,118	No	Storm8	6,314,973	Yes	Freedom Design	6,917,720	No
Dragonplay	3,764,305	Yes	textPlus	5,940,924	Yes	Groupon	6,303,182	Yes	SandStorm Earl	6,611,556	No
Game Insight	3,680,307	Yes	PicPok	5,771,721	Yes	ELT Game Studio	6,234,541	No	Kika Keyboard	6,571,563	No
Dropbox	3,616,981	Yes	Hothead Games	5,736,222	Yes	Etermax	6,150,226	Yes	TutoTOONS	6,522,995	No
Candy Mobile	3,511,685	No	ZXing Team	5,567,719	No	DU Apps	5,986,156	No	Big Duck	6,396,124	Yes
Tiny Piece	3,487,634	No	Backflip Studios	5,430,519	Yes	FT Games	5,877,491	Yes	ZeptoLab	6,379,408	Yes
Tobi	3,462,987	Yes	Bitstrips	5,372,546	Yes	Walmart	5,866,894	Yes	Pretty Simple	6,377,706	Yes
Bejoy Mobile	3,341,855	No	Tumblr	5,339,566	Yes	Big Fish Games	5,811,165	Yes	Mobage	6,373,794	Yes
Kiloo	3,339,662	Yes	AVG	5,286,573	No	Ubisoft	5,800,644	Yes	Holaverse	6,357,277	No
Lucky Art	3,158,792	No	PlayFirst	5,187,822	Yes	RockYou	5,776,511	Yes	eBay	6,038,826	Yes
Autodesk	3,147,183	Yes	KiwiUp	4,897,196	No	Game Show Network	5,633,828	Yes	Activision Publishing	5,941,105	Yes
FreshPlanet	3,095,880	Yes	Bubble Quiz	4,801,767	Yes	KIMSOONgame	5,611,866	No	ooVoo	5,628,984	Yes
Kik	3,080,584	Yes	Zombie Games	4,476,697	No	Mediocre	5,610,350	Yes	Cartoon Network	5,578,936	Yes
PlayScape	3,065,795	No	SoundCloud	4,391,536	Yes	Hothead Games	5,290,369	Yes	AVG	5,432,945	No
Linkedin	3,015,109	Yes	JPMorgan Chase	4,378,829	Yes	Words Mobile	5,039,935	No	TappsGames	5,318,882	Yes
RoidApp	2,702,769	Yes	Klncity1	4,271,230	No	Kika Keyboard	5,029,178	No	Hulu	5,318,729	Yes
DataViz	2,694,693	Yes	Mobile Gaming	4,267,324	No	Candy Mobile	5,007,359	No	Bearbit	5,225,373	Yes
GREE	2,521,439	Yes	Itch Mania	4,106,432	No	Activision Publishing	4,798,433	Yes	Kabam	5,110,505	Yes

Note: A developer is dual-homed if the developer has published top downloaded apps for both Google Play and Apple Store.

Source: "Market Data,"App Annie, data as of December 18, 2015.

Exhibit 2f
Top Developers Ranked by Paid App Downloads

2012-2015

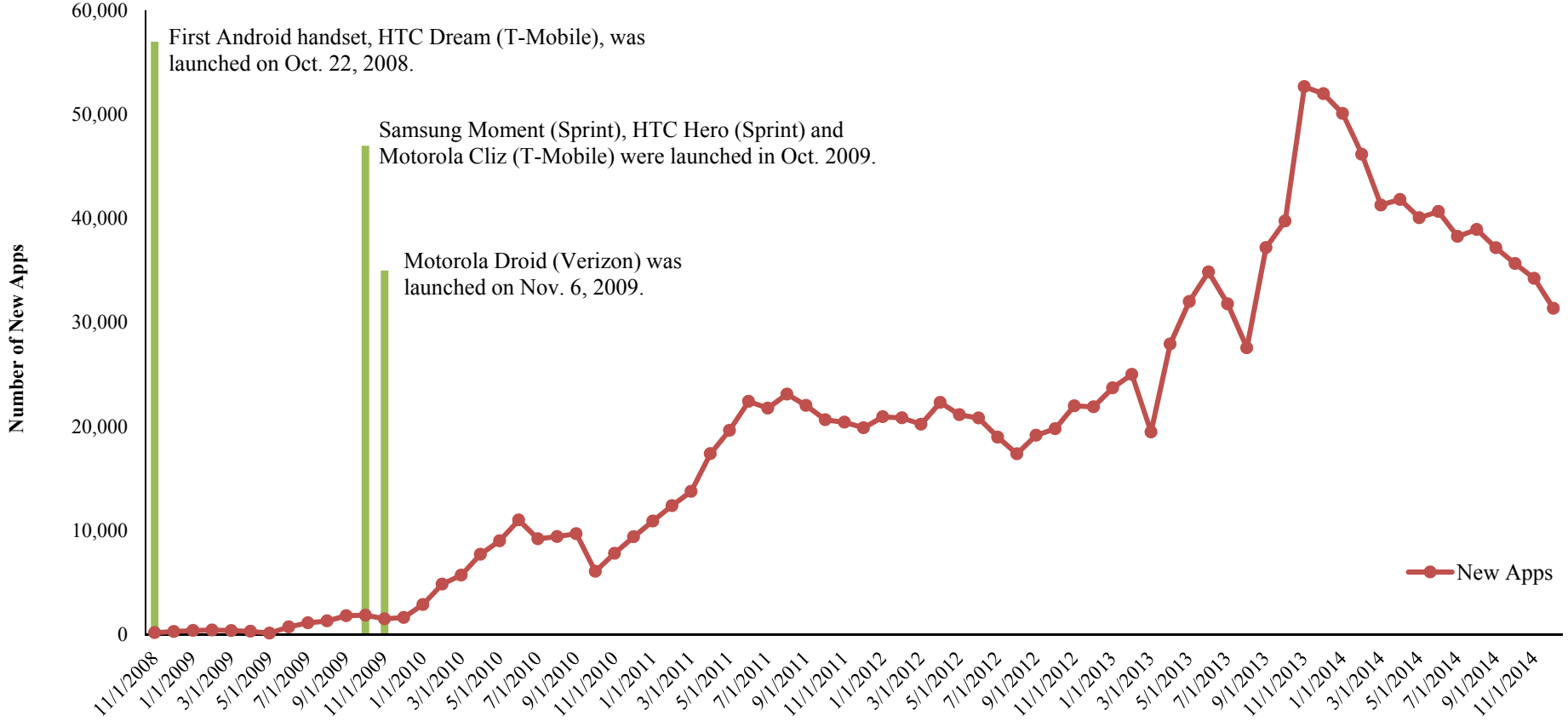
2012			2013			2014			2015		
Developers	**Downloads**	**Dual-Homed**	**Developers**	**Downloads**	**Dual-Homed**	**Developers**	**Downloads**	**Dual-Homed**	**Developers**	**Downloads**	**Dual-Homed**
Electronic Arts	1,313,219	Yes	Electronic Arts	1,541,077	Yes	Electronic Arts	1,674,668	Yes	Mojang	1,049,021	Yes
Gameloft	1,092,970	Yes	Disney	1,242,863	Yes	Mojang	1,544,418	Yes	Electronic Arts	719,176	Yes
Disney	1,081,915	Yes	Mojang	1,000,499	Yes	Disney	583,130	Yes	Scottgames	697,284	Yes
ZeptoLab	760,387	Yes	Gameloft	784,197	Yes	Gameloft	579,983	Yes	SQUARE ENIX	412,320	Yes
OMGPOP	646,508	Yes	SwiftKey	721,929	Yes	Rockstar Games	386,400	Yes	Warner Bros	393,896	Yes
SwiftKey	555,081	Yes	Titanium Track	456,604	No	Scottgames	360,986	Yes	RobTop	332,692	Yes
LevelUpStudio	490,178	No	TeslaCoil	445,335	No	Cartoon Network	358,345	Yes	Cartoon Network	328,779	Yes
Rovio	489,765	Yes	Rockstar Games	420,676	Yes	Ninja Kiwi	308,285	Yes	Rockstar Games	321,694	Yes
Full Fat	449,223	Yes	Rovio	347,868	Yes	Activision Publishing	293,249	Yes	Ninja Kiwi	310,252	Yes
Kittehface	379,920	Yes	Speed Software	346,059	Yes	Fireproof Games	290,381	Yes	Gluten Free Games	253,619	Yes
Halfbrick	373,275	Yes	SEGA	322,333	Yes	TeslaCoil	288,815	No	505 Games	250,190	Yes
Mojang	370,533	Yes	ClockworkMod	315,534	No	SQUARE ENIX	284,569	Yes	Nickelodeon	249,486	Yes
Titanium Track	366,229	No	ZeptoLab	313,633	Yes	Warner Bros	283,110	Yes	Coffee Stain	241,428	Yes
Rockstar Games	364,893	Yes	LevelUpStudio	287,400	No	Max MP	242,309	No	Firemint	226,315	Yes
GameHouse	360,736	Yes	Kittehface	287,179	Yes	SEGA	228,211	Yes	Disney	219,642	Yes
Speed Software	341,114	Yes	Max MP	280,442	No	FoxFi	222,701	No	Etermax	206,333	Yes
Androidslide	299,534	No	Chainfire	272,226	No	ClockworkMod	219,178	No	Big Fish Games	192,870	Yes
ClockworkMod	265,091	No	Halfbrick	267,241	Yes	505 Games	213,776	Yes	Fireproof Games	192,330	Yes
TeslaCoil	264,039	No	MAG Interactive	263,082	Yes	Nickelodeon	204,202	Yes	ustwo	184,601	Yes
Hemisphere Games	262,689	Yes	JRummy Apps	260,016	No	Titanium Track	200,780	No	Gameloft	170,458	Yes
Max MP	261,600	No	CloudTV	240,601	No	Candy Rufus	198,790	Yes	True Axis	166,652	Yes
Outfit7	259,027	Yes	FoxFi	240,162	No	Ubisoft	191,818	Yes	Toca Boca	163,859	Yes
TuneIn	251,577	Yes	Ninja Kiwi	217,530	Yes	Bubble Zap	184,966	No	Ultimate Guitar	156,946	Yes
MADFINGER	241,982	Yes	Androidslide	212,759	No	SwiftKey	175,545	Yes	Activision Publishing	146,585	Yes
2D Boy	234,767	Yes	Quoord	204,045	Yes	Halfbrick	164,034	Yes	HotSchedules	133,548	Yes
JFDP Labs	225,203	Yes	Full Fat	187,706	Yes	Chainfire	157,334	No	Flipline	130,581	Yes
Freedom Design	221,279	No	SQUARE ENIX	184,628	Yes	Ironhide	153,515	Yes	TeslaCoil	129,153	No
Polarbit	203,640	Yes	MobiSystems	179,227	Yes	Ultimate Guitar	150,699	Yes	Telltale Games	124,559	Yes
Soundhound	201,335	Yes	Zynga	176,079	Yes	Plexapp	147,222	Yes	SEGA	116,370	Yes
JRummy Apps	194,294	No	Android Does	174,731	No	Speed Software	140,372	Yes	Halfbrick	111,087	Yes
Mediocre	190,443	Yes	TuneIn	169,871	Yes	Kittehface	138,122	Yes	ClockworkMod	110,308	No
Kairosoft	188,038	Yes	XDA Developers	168,163	No	Coffee Stain	134,192	Yes	ShiftyJelly	108,552	Yes
CloudTV	187,645	No	Activision Publishing	167,570	Yes	Nuance	133,740	Yes	Aspyr	106,632	Yes
Zynga	187,576	Yes	Bubble Zap	167,283	No	Team17	131,688	Yes	Kairosoft	103,842	Yes
Com2uS	186,894	Yes	Hemisphere Games	159,462	Yes	Crafty Apps EU	129,440	No	2K Games	103,378	Yes
Oceanhouse Media	185,513	Yes	Smart Tools	157,378	No	RobTop	128,517	Yes	Ironhide	103,247	Yes
SEGA	185,395	Yes	Mediocre	153,380	Yes	CloudTV	124,945	No	Lightricks	98,971	Yes
Quoord	180,524	Yes	Kairosoft	152,808	Yes	Ian Hanwkins	124,291	No	PuffballsUnited	97,263	Yes
DataViz	173,823	Yes	Crafty Apps EU	151,191	No	Rovio	122,466	Yes	Giants Software	96,442	Yes
Rubicon Development	166,819	Yes	Earth Networks	146,398	Yes	Kairosoft	118,743	Yes	Sony Pictures	95,383	Yes
Latedroid	166,152	No	Oceanhouse Media	146,120	Yes	Magmic	117,162	Yes	Chillingo	93,934	Yes
Monotype Imaging	159,387	No	Nuance	144,948	Yes	LevelUpStudio	111,297	No	PBS KIDS	92,776	Yes
Shazam Entertainment	157,461	Yes	2D Boy	144,723	Yes	JRummy Apps	110,898	No	Candy Rufus	89,503	Yes
DualBoot	150,848	No	Aviary	138,406	Yes	2K Games	110,439	Yes	Ian Hanwkins	89,007	No
Smart Tools	144,313	No	Soundhound	138,216	Yes	True Axis	110,230	Yes	Northcube	88,552	Yes
Shinycore	143,849	No	DualBoot	136,737	No	GO Dev Team	108,198	No	Team17	87,595	Yes
Earth Networks	142,813	Yes	Candy Rufus	134,064	Yes	HotSchedules	107,217	Yes	Crafty Apps EU	83,076	No
XDA Developers	139,796	No	Vector Unit	133,219	Yes	MobiSystems	106,869	Yes	Apalon	81,688	Yes
MichaelHuang	139,089	No	Ian Hanwkins	130,080	No	ZeptoLab	106,691	Yes	FoxFi	78,434	No
DoubleTwist	136,171	No	Shinycore	127,111	No	Androidslide	103,969	No	Epsxe	77,992	No
Lupis	134,481	No	Autodesk	123,850	Yes	Wolfram Alpha	100,498	Yes	ilmfinity	77,983	No
Vector Unit	131,769	Yes	Rerware	121,886	No	Sony Pictures	93,491	Yes	Max MP	76,867	No
Unidocs	130,108	Yes	Glu	121,070	Yes	TuneIn	90,812	Yes	Androidslide	74,280	No
Autodesk	124,564	Yes	DataViz	118,483	Yes	Smart Tools	87,882	No	Stealthychief	70,096	No
Rerware	122,324	No	Latedroid	116,643	No	AccuWeather	84,117	Yes	Gamesoul Studio	69,230	Yes
intellijoy	122,144	No	Shazam Entertainment	115,650	Yes	FlukeDude	80,902	Yes	Nuance	68,299	Yes
Williams Interactive	120,227	Yes	MichaelHuang	115,649	No	Hemisphere Games	78,836	Yes	Tribeplay	65,357	Yes
AnderWeb	115,644	No	DoubleTwist	115,366	No	Sirvo	78,232	Yes	DreamSky	65,266	Yes
Adult Swim	112,029	Yes	Ultimate Guitar	114,271	Yes	ShiftyJelly	76,143	Yes	Wolfram Alpha	64,372	Yes
Dasur	111,240	No	Maxelus	113,263	No	Toca Boca	74,254	Yes	Playdead	62,179	Yes
Android Does	107,427	No	JFDP Labs	112,298	Yes	AI TYPE	73,735	No	HandyGames	62,119	Yes
Magmic	105,262	Yes	OMGPOP	111,062	Yes	IntSig	71,591	Yes	Urbandroid Team	60,563	No
Infinite Dreams	102,827	Yes	Monotype Imaging	109,594	No	Runtastic	70,020	Yes	Kittehface	59,615	Yes
Camel Games	101,816	Yes	Ubisoft	103,403	Yes	Shinycore	69,864	No	Bossa	59,588	Yes
Mobiata	101,745	No	Plexapp	101,290	Yes	Soundhound	69,526	Yes	Scholly	57,204	Yes
MobiSystems	100,135	Yes	Unidocs	98,372	Yes	BoltCreative	68,529	Yes	Runtastic	56,963	Yes
Chainfire	98,511	No	Wolfram Alpha	96,063	Yes	intellijoy	67,878	No	Zuhanden	54,512	No
Ultimate Guitar	97,535	Yes	GO Dev Team	95,140	No	Giants Software	65,592	Yes	Vertumus	53,518	No
BANDAI NAMCO Ente	97,397	Yes	intellijoy	95,053	No	Rerware	65,289	No	Zen Studios	51,658	Yes
GAMEVIL	93,939	Yes	Magmic	85,318	Yes	Oceanhouse Media	62,582	Yes	INXILE	50,430	Yes
Endomondo.com	90,634	Yes	Fast Emulator	84,029	No	Big Fish Games	62,288	Yes	TeamSpeak	50,409	No

2012			2013			2014			2015		
Developers	**Downloads**	**Dual-Homed**	**Developers**	**Downloads**	**Dual-Homed**	**Developers**	**Downloads**	**Dual-Homed**	**Developers**	**Downloads**	**Dual-Homed**
Quickoffice	88,094	No	Quartic	83,510	No	ustwo	62,225	Yes	Outerminds	49,895	Yes
SQUARE ENIX	81,403	Yes	Mobiata	80,743	No	Shazam Entertainment	61,254	Yes	Magmic	49,871	Yes
MLB	77,834	Yes	AccuWeather	80,460	Yes	Vector Unit	61,188	Yes	Nekki	48,517	Yes
SPB Software	77,641	No	JRTStudio	79,943	No	1337 and Senri	61,138	Yes	Ubisoft	48,412	Yes
DistinctDev	75,020	Yes	Adult Swim	79,786	Yes	Autodesk	59,059	Yes	Noodlecake Studios	47,884	Yes
Bithack	75,004	No	ShiftyJelly	77,556	Yes	Full Fat	58,655	Yes	Bltstudlo	46,473	No
Ian Hanwkins	73,485	No	snrb Labs	76,780	No	JRTStudio	57,753	No	Reliance Big Entertainmer	45,835	Yes
Bubble Zap	71,533	No	Williams Interactive	76,494	Yes	Not Doppler	53,293	Yes	Dude Perfect	45,158	Yes
BeansoftApps	69,730	No	Fireproof Games	72,509	Yes	Flipline	53,050	Yes	JRTStudio	43,666	No
Crafty Apps EU	69,421	No	Endomondo.com	72,392	Yes	DoubleTwist	52,709	No	Simon Filip	43,446	Yes
BoltCreative	69,389	Yes	AnderWeb	64,796	No	Monotype Imaging	52,204	No	EightyEight Games	42,644	Yes
Meltus	67,433	No	Escargot	60,717	No	Urbandroid Team	51,270	No	RichFace	40,290	No
NitroDesk	66,983	No	Runtastic	60,597	Yes	Crescent Moon Games	50,798	Yes	gravity sensation	39,607	Yes
FDP	60,180	No	GameHouse	60,453	Yes	DataViz	50,205	Yes	Sam Ruston	39,540	No
Appgenix	58,980	No	Warner Bros	58,594	Yes	Gluten Free Games	49,613	Yes	SNK PLAYMORE	38,642	Yes
Glu	58,335	Yes	Joaquim Vergès	57,516	No	Northcube	49,229	Yes	ActiveNetwork	37,172	Yes
Defiant Development	57,355	Yes	PlayFirst	56,634	Yes	JFDP Labs	48,766	Yes	1337 and Senri	37,109	Yes
Escapist	57,164	No	Ironhide	54,078	Yes	XDA Developers	48,728	No	BoltCreative	37,020	Yes
Edward Kim	55,185	No	Dynamix Software	52,948	No	INXILE	48,297	Yes	Not Doppler	34,245	Yes
Maxelus	53,700	No	Limbic	52,506	Yes	Mediocre	46,372	Yes	Chainfire	30,980	No
Splashtop	51,897	Yes	Kovdev	51,622	No	Etermax	45,512	Yes	Zhuowei Zhang	30,878	No
IntSig	50,443	Yes	BlastOn	51,225	No	Glu	45,453	Yes	USMGames	30,709	Yes
Fallen Tree Games	48,358	No	DPFlashes	50,990	Yes	Dynamix Software	44,394	No	Imagine Ink	29,578	No
HeroCraft	47,947	Yes	Rubicon Development	50,771	Yes	Kovdev	43,294	No	Alawar	29,411	Yes
ChainsDD	45,652	No	Mathias Roth	50,488	No	Epsxe	43,235	No	Days Of Wonder	28,536	Yes
SVOX	45,242	No	Outfit7	49,260	Yes	Jundroo	41,759	Yes	intellijoy	28,095	No
Dexilog	43,773	No	Defiant Development	45,629	Yes	Sling Media	41,744	Yes	WeatherSphere	27,958	Yes
JRTStudio	43,756	No	Quickoffice	42,313	No	Noodlecake Studios	40,680	Yes	Crescent Moon Games	27,829	Yes
BlastOn	43,077	No	Crescent Moon Games	41,501	Yes	Simon Filip	38,462	Yes	Joel McDonald	26,393	Yes

Note: A developer is dual-homed if the developer has published top downloaded apps for both Google Play and Apple Store.

Source: "Market Data,"App Annie, data as of December 18, 2015.

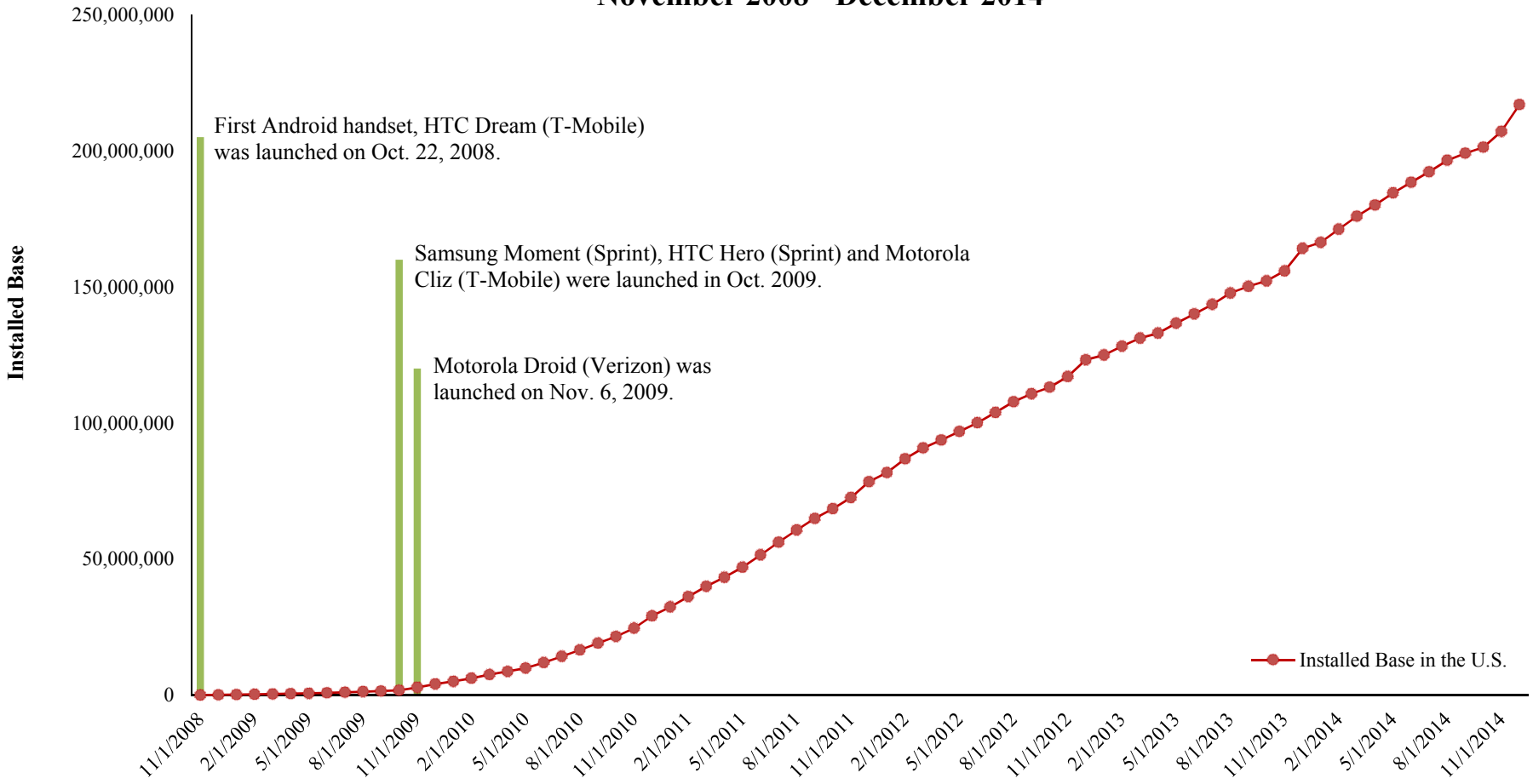
Exhibit 2g
Number of New Apps on Google Play
November 2008 - December 2014



Notes: Total numbers of apps available on Google Play on the first day of each month between Nov. 2008 and Dec. 2014 are estimated using non-parametric imputation. New apps are calculated as the increase in the total number of apps at monthly level.

Sources: "Applications," Androlib.com.
"Number of Android Applications," AppBrain.com.
"A Brief History of Android Phones," CNET, August 2, 2011, http://www.cnet.com/news/a-brief-history-of-android-phones/.
"Motorola 'Droid Does' Release: First Android 2.0 Phone Unveiled, Price Revealed (Updated, Photos)," Huffington Post, May 25, 2011, http://www.huffingtonpost.com/2009/10/22/motorola-droid-does-relea_n_330100.html.
"T-Mobile G1 Details, Price, and Launch Date Revealed," CNET, October 16, 2008, http://www.cnet.com/news/t-mobile-g1-details-price-and-launch-date-revealed/.

Exhibit 2h
Installed Base of Android Smartphones
November 2008 - December 2014



Notes: Installed base in month t is defined as the sum of activations in month t, and the installed base in month t-1, minus deactivations.
Activated users are assumed to deactivate evenly over five months after 22 months.

Sources: GOOG-00022382.
"A Brief History of Android Phones," CNET, August 2, 2011, http://www.cnet.com/news/a-brief-history-of-android-phones/.
"Motorola 'Droid Does' Release: First Android 2.0 Phone Unveiled, Price Revealed (Updated, Photos)," Huffington Post, May 25, 2011, http://www.huffingtonpost.com/2009/10/22/motorola-droid-does-relea_n_330100.html.
"T-Mobile G1 Details, Price, and Launch Date Revealed," CNET, October 16, 2008, http://www.cnet.com/news/t-mobile-g1-details-price-and-launch-date-revealed/.

Exhibit 2i
Free and Paid Apps Appearing on Daily Top 100 Download Lists

June 2015

App ID (a)	App Name (b)	Developer Name (c)	Type (d)	Monthly App Downloads (e)	App is Dual-Homed (f)	Google App (g)	C++ App (h)	Developer is Dual-Homed (i)	Developer Wrote C++ Apps (j)	Microsft Counterpart Available (k)
com.facebook.orca	Messenger	Facebook	Free	4,566,815	Yes	No	Yes	Yes	Yes	--
com.facebook.katana	Facebook	Facebook	Free	3,207,894	Yes	No	No	Yes	Yes	--
com.google.android.apps.photos	Google Photos	Google Inc.	Free	3,079,842	Yes	Yes	No	Yes	Yes	--
com.pandora.android	Pandora® Radio	Pandora	Free	2,680,432	Yes	No	No	Yes	No	--
com.king.alphabettysaga	AlphaBetty Saga	King	Free	2,569,230	Yes	No	No	Yes	Yes	--
com.instagram.android	Instagram	Instagram	Free	2,448,535	Yes	No	Yes	Yes	Yes	--
com.snapchat.android	Snapchat	Snapchat Inc	Free	2,230,163	Yes	No	Yes	Yes	Yes	--
com.surpax.ledflashlight.panel	Super-Bright LED Flashlight	Surpax Inc.	Free	2,095,809	Yes	No	No	Yes	No	--
com.netflix.mediaclient	Netflix	Netflix, Inc.	Free	1,935,350	Yes	No	Yes	Yes	Yes	--
com.spotify.music	Spotify Music	Spotify Ltd.	Free	1,871,892	Yes	No	No	Yes	No	--
kik.android	Kik	Kik Interactive	Free	1,701,154	Yes	No	Yes	Yes	Yes	--
com.miniclip.dudeperfect	Dude Perfect 2	Miniclip.com	Free	1,678,364	Yes	No	No	Yes	Yes	--
com.whatsapp	WhatsApp Messenger	WhatsApp Inc.	Free	1,611,144	Yes	No	Yes	Yes	Yes	--
com.google.android.apps.inbox	Inbox by Gmail	Google Inc.	Free	1,538,664	Yes	Yes	No	Yes	Yes	--
com.yodo1.crossyroad	Crossy Road	Yodo1 Games	Free	1,516,435	Yes	No	No	Yes	No	--
com.mobilemotion.dubsmash	Dubsmash	Mobile Motion GmbH	Free	1,500,111	Yes	No	No	Yes	No	--
com.kiloo.subwaysurf	Subway Surfers	Kiloo	Free	1,471,720	Yes	No	Yes	Yes	Yes	--
com.instagram.layout	Layout from Instagram: Collage	Instagram	Free	1,461,537	Yes	No	No	Yes	Yes	--
com.cleanmaster.mguard	Clean Master (Boost & AppLock)	Cheetah Mobile	Free	1,461,046	No	No	No	Yes	No	--
com.skype.raider	Skype - free IM & video calls	Skype	Free	1,416,305	Yes	No	No	Yes	No	--
com.qihoo.security	360 Security - Antivirus Boost	360 Mobile Security Limited	Free	1,403,071	No	No	No	No	No	Yes
fr.x_studios.x_laser_2	Laser Pointer X2 Simulator	X-Apps & X-Games	Free	1,372,072	No	No	No	No	No	No
com.king.candycrushsodasaga	Candy Crush Soda Saga	King	Free	1,369,497	Yes	No	Yes	Yes	Yes	--
com.prettysimple.criminalcaseandroid	Criminal Case	Pretty Simple	Free	1,344,952	Yes	No	No	Yes	No	--
com.miniclip.eightballpool	8 Ball Pool	Miniclip.com	Free	1,277,469	Yes	No	No	Yes	Yes	--
com.pinterest	Pinterest	Pinterest, Inc.	Free	1,266,858	Yes	No	Yes	Yes	Yes	--
com.supercell.clashofclans	Clash of Clans	Supercell	Free	1,260,596	Yes	No	Yes	Yes	Yes	--
com.amazon.mShop.android.shopping	Amazon Shopping	Amazon Mobile LLC	Free	1,215,773	Yes	No	No	Yes	No	--
com.king.candycrushsaga	Candy Crush Saga	King	Free	1,178,446	Yes	No	Yes	Yes	Yes	--
net.zedge.android	ZEDGE™ Ringtones & Wallpapers	Zedge	Free	1,174,971	Yes	No	No	Yes	No	--
com.nordcurrent.canteenhd	Cooking Fever	Nordcurrent	Free	1,110,031	Yes	No	No	Yes	No	--
com.emoji.coolkeyboard	Emoji Keyboard Pro Kika Free	Kika Keyboard Tech	Free	1,054,680	No	No	No	No	No	Yes
com.imangi.templerun2	Temple Run 2	Imangi Studios	Free	1,046,515	Yes	No	Yes	Yes	Yes	--
com.soundcloud.android	SoundCloud - Music & Audio	SoundCloud	Free	1,042,902	Yes	No	Yes	Yes	Yes	--
com.gameloft.android.ANMP.GloftDMHM	Despicable Me	Gameloft	Free	1,023,725	Yes	No	No	Yes	Yes	--
com.weather.Weather	The Weather Channel	The Weather Channel	Free	966,965	Yes	No	No	Yes	No	--
com.google.android.play.games	Google Play Games	Google Inc.	Free	960,452	No	Yes	No	Yes	Yes	--
com.twitter.android	Twitter	Twitter, Inc.	Free	937,618	Yes	No	Yes	Yes	Yes	--
com.lazyswipe	Omni Swipe-Small,Quick	Holaverse Group	Free	909,374	No	No	No	No	No	No
com.yahoo.mobile.client.android.mail	Yahoo Mail – Free Email App	Yahoo	Free	897,293	Yes	No	No	Yes	No	--
com.hcg.cok.gp	Clash of Kings	Elex Wireless	Free	887,466	Yes	No	No	Yes	No	--

App ID (a)	App Name (b)	Developer Name (c)	Type (d)	Monthly App Downloads (e)	App is Dual-Homed (f)	Google App (g)	C++ App (h)	Developer is Dual-Homed (i)	Developer Wrote C++ Apps (j)	Microsft Counterpart Available (k)
com.contextlogic.wish	Wish - Shopping Made Fun	Wish Inc.	Free	879,300	Yes	No	No	Yes	No	--
com.kathleenOswald.solitaireGooglePlay	Solitaire	Harpan LLC	Free	877,040	Yes	No	No	Yes	No	--
com.clearchannel.iheartradio.controller	iHeartRadio Free Music & Radio	iHeartMedia, Inc.	Free	851,302	Yes	No	No	Yes	No	--
com.machinezone.gow	Game of War - Fire Age	Machine Zone, Inc.	Free	799,957	Yes	No	No	Yes	No	--
com.wb.goog.mkx	MORTAL KOMBAT X	Warner Bros. International Enterprises	Free	780,936	Yes	No	No	Yes	Yes	--
com.ludia.jurassicworld	Jurassic World™: The Game	Ludia Inc.	Free	778,121	Yes	No	No	Yes	Yes	--
com.ebay.mobile	eBay	eBay Mobile	Free	776,707	Yes	No	Yes	Yes	Yes	--
com.robtopx.geometryjumplite	Geometry Dash Lite	RobTop Games	Free	759,653	Yes	No	No	Yes	No	--
com.musicplayer.music	Default Music Player	JAYAVELU V	Free	752,653	No	No	No	No	No	Yes
com.bigduckgames.flow	Flow Free	Big Duck Games LLC	Free	741,152	Yes	No	Yes	Yes	Yes	--
com.oovoo	ooVoo Video Call, Text & Voice	ooVoo LLC	Free	726,647	Yes	No	No	Yes	No	--
com.sgn.pandapop.gp	Panda Pop	SGN	Free	722,066	Yes	No	No	Yes	No	--
com.igg.android.finalfable	Final Fable	IGG.COM	Free	706,140	No	No	No	Yes	Yes	--
com.bubble.snow2	Bubble Snow	ELT Game Studio	Free	705,787	No	No	No	No	No	Yes
com.ubercab	Uber	Uber Technologies, Inc.	Free	704,892	Yes	No	No	Yes	No	--
com.cheerfulinc.flipagram	Flipagram	Flipagram, inc.	Free	697,016	Yes	No	No	Yes	No	--
tv.twitch.android.app	Twitch	Twitch Interactive, Inc.	Free	674,497	No	No	No	Yes	No	--
com.emojifamily.emoji.keyboard.font.twitteremoji	Emoji Keyboard Cute & Colorful	Emoji Art Design	Free	667,117	No	No	No	No	No	Yes
com.grandegames.slots	Slots Forever™ FREE Casino	Grande Games - Slots and Pokies	Free	659,028	No	No	No	No	No	Yes
com.shazam.android	Shazam	Shazam Entertainment Limited	Free	648,751	Yes	No	Yes	Yes	Yes	--
com.king.farmheroessaga	Farm Heroes Saga	King	Free	641,619	Yes	No	No	Yes	Yes	--
com.google.earth	Google Earth	Google Inc.	Free	640,992	Yes	Yes	Yes	Yes	Yes	--
com.sec.android.easyMover	Samsung Smart Switch Mobile	Samsung Electronics Co., Ltd.	Free	638,009	No	No	No	Yes	No	--
com.sgiggle.production	Tango - Free Video Call & Chat	Tango	Free	635,170	Yes	No	No	Yes	No	--
air.com.sgn.cookiejam.gp	Cookie Jam	SGN	Free	627,957	Yes	No	No	Yes	No	--
com.halfbrick.fruitninjafree	Fruit Ninja Free	Halfbrick Studios	Free	625,083	Yes	No	Yes	Yes	Yes	--
com.umonistudio.tile	Dont Tap The White Tile	Clean Master Games	Free	617,553	Yes	No	No	Yes	No	--
com.cmge.gplay.riseofdarkness	Rise of Darkness	NewGame	Free	616,418	Yes	No	No	Yes	No	No
com.waze	Waze - GPS, Maps & Traffic	Waze	Free	612,099	Yes	No	No	Yes	No	--
com.mgc.miami.crime.simulator	Miami crime simulator	Mine Games Craft	Free	608,948	No	No	No	No	No	No
com.jb.emoji.gokeyboard	GO Keyboard - Emoji, Sticker	GO Keyboard Dev Team	Free	606,413	No	No	No	No	No	Yes
com.wallapop	Wallapop - Buy & Sell Nearby	Wallapop	Free	593,949	Yes	No	No	Yes	No	--
com.ping9games.grabtheauto3	Grab The Auto 3	Ping9 Games	Free	593,406	No	No	No	No	No	No
com.mixradio.droid	MixRadio Stream Free Music	MixRadio Limited	Free	592,723	No	No	No	No	No	Yes
com.bingocrushgame.bingovegas	Bingo - Pro Bingo Crush™	KALE	Free	583,253	No	No	No	No	No	Yes
com.xmatch.cookiestory2	Cookie Story	pixelementgame	Free	577,247	No	No	No	No	No	No
com.fingersoft.hillclimb	Hill Climb Racing	Fingersoft	Free	577,201	Yes	No	Yes	Yes	Yes	--
jp.naver.line.android	LINE: Free Calls & Messages	LINE Corporation	Free	570,648	Yes	No	No	Yes	No	--
com.leftover.CoinDozer	Coin Dozer - Free Prizes	Game Circus LLC	Free	570,449	Yes	No	Yes	Yes	Yes	--
com.offerup	OfferUp - Buy. Sell. Offer Up	OfferUp Inc.	Free	569,632	Yes	No	No	Yes	No	--
co.vine.android	Vine - video entertainment	Vine Labs	Free	568,225	Yes	No	No	Yes	No	--
com.supercell.boombeach	Boom Beach	Supercell	Free	566,946	Yes	No	No	Yes	Yes	--
com.hotheadgames.google.free.rawsniper	Kill Shot	Hothead Games	Free	566,787	Yes	No	No	Yes	No	--
com.earnworld.offercash	QuickCash	MyEarn	Free	563,678	No	No	No	No	No	No
com.etermax.preguntados.lite	Trivia Crack	Etermax	Free	563,595	Yes	No	No	Yes	No	--

App ID (a)	App Name (b)	Developer Name (c)	Type (d)	Monthly App Downloads (e)	App is Dual-Homed (f)	Google App (g)	C++ App (h)	Developer is Dual-Homed (i)	Developer Wrote C++ Apps (j)	Microsft Counterpart Available (k)
com.smule.magicpiano	Magic Piano by Smule	Smule	Free	560,643	Yes	No	Yes	Yes	Yes	--
com.ijinshan.kbatterydoctor_en	Battery Doctor (Battery Saver)	Cheetah Mobile Inc. (NYSE: CMCM)	Free	560,581	Yes	No	No	Yes	No	--
jp.co.ofcr.cm00	COOKING MAMA Lets Cook !	Office Create Corp.	Free	555,234	Yes	No	No	Yes	No	No
com.imangi.templerun	Temple Run	Imangi Studios	Free	549,614	Yes	No	Yes	Yes	Yes	--
com.tinder	Tinder	Tinder	Free	548,885	Yes	No	No	Yes	No	--
com.dianxinos.dxbs	DU Battery Saver&Phone Charger	DU APPS STUDIO	Free	543,675	No	No	No	No	No	Yes
com.hulu.plus	Hulu	Hulu	Free	529,529	Yes	No	No	Yes	No	--
com.microsoft.office.outlook	Microsoft Outlook	Microsoft Corporation	Free	527,419	Yes	No	No	Yes	No	--
com.explorationbase.ExplorationLite	Exploration Lite	Andrzej Chomiak	Free	523,875	Yes	No	No	Yes	No	--
com.ludia.waldo3free	Waldo & Friends	Ludia Inc.	Free	519,614	Yes	No	No	Yes	Yes	--
com.beatsmusic.android.client	Beats Music	Beats Music	Free	519,003	Yes	No	No	Yes	No	--
com.rovio.angrybirds	Angry Birds	Rovio Entertainment Ltd.	Free	518,164	Yes	No	Yes	Yes	Yes	--
com.yelp.android	Yelp	Yelp, Inc	Free	506,295	Yes	No	No	Yes	No	--
com.contextlogic.geek	Geek - Smarter Shopping	Wish Inc.	Free	505,110	No	No	No	Yes	No	--
vu.android	Like Parent	Luong Anh Vu	Free	493,280	No	No	No	No	No	No
com.goldenlion.slots	Golden Lion Slots™-Free Casino	GOLDENCASINO	Free	464,071	No	No	No	No	No	Yes
com.playfirst.cookingdashx	COOKING DASH 2016	Glu	Free	458,466	Yes	No	No	Yes	Yes	--
com.turner.cnblamburger	Blamburger - Clarence	Cartoon Network	Free	381,856	Yes	No	No	Yes	No	--
com.disney.thoughtbubbles_goo	Inside Out Thought Bubbles	Disney	Free	315,519	Yes	No	No	Yes	Yes	--
com.iwin.dond	Deal or No Deal	iWin	Free	295,615	Yes	No	No	Yes	No	--
com.bigkraken.thelastwar	Deadwalk: The Last War	QJ Games	Free	275,909	Yes	No	No	Yes	No	--
com.cleanmaster.security	CM Security Antivirus AppLock	Cheetah Mobile (AppLock & AntiVirus)	Free	263,831	No	No	No	Yes	No	--
com.tap4fun.reignofwar	Invasion: Modern Empire	tap4fun	Free	251,515	Yes	No	No	Yes	No	--
com.umbrella.boomdots	Boom Dots	Mudloop	Free	241,639	Yes	No	No	Yes	No	No
com.ssc.fitfat	Lose Weight - Slimming!	Candy Mobile	Free	241,638	No	No	No	No	No	Yes
br.com.tapps.cowevolution	Cow Evolution - Clicker Game	Tapps - Top Apps and Games	Free	240,074	No	No	No	Yes	No	--
com.FakeCall2	Fake Call 2	Technology expertise	Free	236,469	No	No	No	No	No	No
com.playstudios.myvegas	myVEGAS Slots Free Casino	PlayStudios	Free	213,237	Yes	No	No	Yes	No	--
com.fgol.HungrySharkEvolution	Hungry Shark Evolution	Future Games of London	Free	202,773	Yes	No	Yes	Yes	Yes	--
com.kiwi.skullislandexplorers	Explorers: Skull Island	Rockyou Inc.	Free	201,086	No	No	No	Yes	No	--
frobenius.remote.control.tv	Remote Control for TV	Universal Support	Free	192,053	No	No	No	No	No	Yes
com.fireflygames.rushofheroes	Rush of Heroes	Firefly Games Inc.	Free	189,250	No	No	No	Yes	No	No
com.igg.castleclash	Castle Clash: Age of Legends	IGG.COM	Free	184,948	Yes	No	Yes	Yes	Yes	--
home.solo.launcher.free	Solo Launcher Clean Smooth Diy	Solo Launcher Team	Free	181,190	No	No	No	No	No	No
com.fungames.lords	Infinite Myths 2: Soul Lords	2MuchFun	Free	180,304	No	No	No	No	No	No
air.com.differencegames.hiddensce nes.countrycornerfree	Hidden Scenes - Country Corner	Difference Games LLC	Free	173,979	No	No	No	Yes	No	No
com.slots.classicvegas	Classic Vegas Slots	Exotron	Free	169,962	No	No	No	No	No	Yes
emoji.keyboard.emoticonkeyboard	Emoji Keyboard Cute Emoticons	Colorful Emoji Keyboard	Free	161,765	No	No	No	No	No	Yes
com.smule.singandroid	Sing! Karaoke by Smule	Smule	Free	158,261	Yes	No	No	Yes	Yes	--
my.googlemusic.play	My Mixtapez Music & Mixtapes	My Mixtapez LLC.	Free	156,521	Yes	No	No	Yes	No	--
com.eliferun.music	Music Player for Android	turtlerun	Free	154,168	No	No	No	No	No	Yes
com.aim.racing	Extreme Car Driving Simulator	AxesInMotion Racing	Free	152,784	Yes	No	No	Yes	No	--
multicraft.worldcraft.world	Exploration WorldCraft 2 PE	Best Free App Studio	Free	150,852	No	No	No	No	No	--
com.madebyappolis.spinrilla	Spinrilla	Spinrilla	Free	149,732	Yes	No	No	Yes	No	--
com.timgames.findthedifferencetim2	Find Differences	timoffstudio	Free	147,333	No	No	No	No	No	Yes
com.melodis.midomiMusicIdentifie r.freemium	SoundHound Music Search	SoundHound Inc.	Free	147,147	Yes	No	Yes	Yes	Yes	--
com.uken.BingoPop	Bingo Pop	Uken Games	Free	146,969	Yes	No	No	Yes	No	--

App ID (a)	App Name (b)	Developer Name (c)	Type (d)	Monthly App Downloads (e)	App is Dual-Homed (f)	Google App (g)	C++ App (h)	Developer is Dual-Homed (i)	Developer Wrote C++ Apps (j)	Microsft Counterpart Available (k)
com.huuuge.zeus	Slots - Zeus Casino Games	HUUUGE GAMES	Free	146,178	No	No	No	Yes	No	--
com.notdoppler.earntodie2	Earn to Die 2	Not Doppler	Free	145,166	Yes	No	Yes	Yes	Yes	--
com.nextradioapp.nextradio	NextRadio - Free Live FM Radio	Next Radio, LLC	Free	144,834	No	No	No	No	No	Yes
tunein.player	TuneIn Radio - Radio & Music	TuneIn Inc	Free	143,217	Yes	No	No	Yes	No	--
com.igg.bzbee.slotsdeluxe	Slot Machines by IGG	IGG.COM	Free	142,818	Yes	No	No	Yes	Yes	--
com.igg.clashoflords2	Clash of Lords 2: Heroes War	IGG.COM	Free	137,227	Yes	No	No	Yes	Yes	--
com.outplayentertainment.bubblebl aze	Bubble Blaze	Outplay Entertainment Ltd	Free	135,769	Yes	No	No	Yes	No	--
com.emoji.keyboard.touchpal	TouchPal Emoji Keyboard	TouchPal Emoji Keyboard Team	Free	131,664	Yes	No	No	Yes	No	--
com.jrtstudio.music	Music Player	JRT Studio	Free	131,110	No	No	No	No	No	Yes
com.z14.darkdemon	Dark of the Demons	Z14 GAMES	Free	131,016	Yes	No	No	Yes	No	No
media.music.musicplayer	Music Player - Audio Player	Mobile_V5	Free	130,806	No	No	No	No	No	Yes
com.dg.gamgogames.venicemystery	Mahjong Venice Mystery Puzzle	Difference Games LLC	Free	130,221	No	No	No	Yes	No	--
com.zentertain.photoeditor	Photo Editor Pro	Zentertain	Free	125,472	Yes	No	No	Yes	No	--
hr.podlanica	Music Volume EQ	K&K design	Free	124,423	No	No	No	No	No	No
com.igg.pokerdeluxe	Texas HoldEm Poker Deluxe	IGG.COM	Free	124,351	Yes	No	No	Yes	Yes	--
com.rhapsody	Rhapsody Music Player	Rhapsody International, Inc.	Free	123,792	Yes	No	No	Yes	No	--
com.zynga.farmarcade	FarmVille: Harvest Swap	Zynga	Free	122,951	Yes	No	No	Yes	No	--
com.blizzard.wtcg.hearthstone	Hearthstone Heroes of Warcraft	Blizzard Entertainment, Inc.	Free	122,948	Yes	No	No	Yes	No	--
com.gameone.fruit.revels	Fruit Revels	gameone	Free	121,172	No	No	No	No	No	--
air.com.fgl.happyplanetgames.cube crash2deluxefree	Cube Crash 2 Deluxe Free	Ocean Breeze Games	Free	120,546	No	No	No	No	No	--
com.rovio.angrybirdsfight	Angry Birds Fight! RPG Puzzle	Rovio Entertainment Ltd.	Free	119,948	Yes	No	Yes	Yes	Yes	--
com.picsart.studio	PicsArt Photo Studio	PicsArt	Free	114,017	Yes	No	No	Yes	No	--
com.zentertain.classicvegasslots	Slots™ - Classic Vegas Casino	ZENTERTAIN LTD	Free	110,243	Yes	No	No	Yes	No	--
com.eRepublikLabs.AgeOfLords	Age of Lords: Legends & Rebels	Erepublik Labs	Free	107,052	Yes	No	No	Yes	No	No
dk.tactile.beebrilliant	Bee Brilliant	Tactile Entertainment	Free	106,190	Yes	No	No	Yes	No	--
com.zentertain.bigcasino	Lucky Win Casino™- FREE SLOTS	ZENTERTAIN LTD	Free	105,474	No	No	No	Yes	No	--
air.com.sgn.juicejam.gp	Juice Jam	SGN	Free	100,103	Yes	No	No	Yes	No	--
com.digitalsky.dragonbane.elite	Dragon Bane Elite	Digital Sky Entertainment Ltd.	Free	100,013	Yes	No	No	Yes	No	--
com.fivestargames.slots	Slots Free - Big Win Casino™	FiveStar Games - Slots and Casino	Free	98,271	No	No	No	No	No	Yes
com.mojang.minecraftpe	Minecraft: Pocket Edition	Mojang	Paid	95,540	Yes	No	Yes	Yes	Yes	--
com.tp.android.plasticsurgery	Plastic Surgery Simulator	6677g.com	Free	90,364	No	No	No	Yes	Yes	--
com.mobilityware.solitaire	Solitaire	MobilityWare	Free	89,926	Yes	No	No	Yes	No	--
com.gamelion.cats	Cats Dogs Slots&Slot machines	HUUUGE GAMES	Free	88,940	No	No	No	Yes	No	--
com.glu.deerhunt2	DEER HUNTER 2014	Glu	Free	88,212	Yes	No	No	Yes	Yes	--
com.econdevpros.games.machunter	Mad Swing Rider Free	MadSkill Game Studios, LLC.	Free	74,094	No	No	No	No	No	--
com.sm.smove	Smove	Simple Machine	Free	70,093	Yes	No	No	Yes	No	--
air.com.flowplay.FringoMobile	Fringo Bingo & Slots Together	FlowPlay, Inc.	Free	69,887	No	No	No	No	No	Yes
tv.periscope.android	Periscope	Twitter, Inc.	Free	64,033	Yes	No	No	Yes	Yes	--
org.mozilla.firefox	Firefox Browser for Android	Mozilla	Free	61,419	Yes	No	No	Yes	No	--
com.ciegames.RacingRivals	Racing Rivals	Glu	Free	60,162	Yes	No	No	Yes	Yes	--
media.mp3player.musicplayer	Music Player - Mp3 Player	Easyelife	Free	59,992	No	No	No	No	No	--
com.acmeaom.android.myradar	MyRadar Weather Radar	ACME AtronOmatic	Free	59,109	Yes	No	No	Yes	No	--
com.apusapps.launcher	APUS Launcher-Small,Fast,Boost	Apus Group	Free	57,602	No	No	No	No	No	No
com.amazon.mp3	Amazon Music with Prime Music	Amazon Mobile LLC	Free	57,388	Yes	No	No	Yes	No	--
com.cmcm.locker	CM Locker (Secure & Boost)	Cheetah Mobile (Secure Lockscreen)	Free	56,554	No	No	No	Yes	No	--
com.igg.bzbee.deckheroes	Deck Heroes: Legacy	IGG.COM	Free	56,024	No	No	No	Yes	Yes	--

App ID (a)	App Name (b)	Developer Name (c)	Type (d)	Monthly App Downloads (e)	App is Dual-Homed (f)	Google App (g)	C++ App (h)	Developer is Dual-Homed (i)	Developer Wrote C++ Apps (j)	Microsft Counterpart Available (k)
com.teamlava.restaurantstory46	Restaurant Story: Summer Camp	Storm8 Studios	Free	55,857	No	No	Yes	Yes	Yes	--
com.ea.game.tetris2011_na	TETRIS®	Electronic Arts Inc	Free	55,271	Yes	No	No	Yes	Yes	--
com.outfit7.mytalkingtomfree	My Talking Tom	Outfit7	Free	54,533	Yes	No	Yes	Yes	Yes	--
com.microsoft.office.word	Microsoft Word	Microsoft Corporation	Free	50,412	Yes	No	No	Yes	No	--
com.boombit.RunningCircles	Running Circles	BoomBit Games	Free	48,394	No	No	No	Yes	No	--
com.madskillgames.games.defendourworld	Defend Our World	Tenth Street Studios	Free	45,711	No	No	No	No	No	--
air.com.puffballsunited.escapingtheprison	Escaping the Prison	PuffballsUnited	Free	43,547	Yes	No	No	Yes	No	No
com.oceanview.twenty48reborn	2048 Reborn	Oceanview Games	Free	42,638	No	No	No	No	No	--
com.alis.hero_uncube	Need A Hero: Princess Rescue	Alis Games	Free	40,336	Yes	No	No	Yes	No	--
com.tinyco.familyguy	Family Guy The Quest for Stuff	TinyCo	Free	39,654	Yes	No	No	Yes	Yes	--
com.forgame.battle	Battle for Domination	FORGAME	Free	36,770	No	No	No	No	No	--
com.sirma.mobile.bible.android	Bible	Life.Church	Free	36,274	Yes	No	No	Yes	No	--
com.spaceapegames.rivalkingdoms	Rival Kingdoms: Age of Ruin	Space Ape Games	Free	30,880	Yes	No	No	Yes	Yes	--
com.ninjakiwi.bloonstd5	Bloons TD 5	ninja kiwi	Paid	29,076	Yes	No	Yes	Yes	Yes	--
com.robtopx.geometryjump	Geometry Dash	RobTop Games	Paid	27,023	Yes	No	No	Yes	No	--
org.pbskids.dtgrrificfeelings	Daniel Tiger Grr-ific Feelings	PBS KIDS	Paid	25,237	Yes	No	No	Yes	No	--
com.perblue.greedforglory	Greed for Glory: War Strategy	PerBlue	Free	24,189	No	No	No	Yes	No	--
au.com.shiftyjelly.pocketcasts	Pocket Casts	Shifty Jelly	Paid	23,636	Yes	No	No	Yes	No	No
net.peakgames.amy	Toy Blast	Peak Games	Free	22,610	Yes	No	No	Yes	No	--
com.wb.goog.scribbleremix	Scribblenauts Remix	Warner Bros. International Enterprises	Paid	22,290	Yes	No	Yes	Yes	Yes	--
com.eightyeightgames.ymbab	You Must Build A Boat	EightyEight Games	Paid	22,115	Yes	No	No	Yes	No	--
com.shakespeare.slots.android	SLOTS: Shakespeare Slot Games!	Super Lucky Casino: Free Slot Machines Bingo Games	Free	21,321	No	No	No	Yes	No	--
com.eamobile.life_na_wf	THE GAME OF LIFE	Electronic Arts Inc	Paid	20,780	Yes	No	No	Yes	Yes	--
com.ea.games.simsfreeplay_na	The Sims™ FreePlay	ELECTRONIC ARTS	Free	18,825	Yes	No	Yes	Yes	Yes	--
com.roostergames.hillclimbtruckracing3	Truck Driver 3D: Offroad	Rooster Games	Free	18,699	No	No	No	No	No	No
com.glu.t5	TERMINATOR GENISYS: GUARDIAN	Glu	Free	18,660	Yes	No	No	Yes	Yes	--
com.scottgames.fivenightsatfreddys	Five Nights at Freddys	Scott Cawthon	Paid	18,621	Yes	No	No	Yes	No	--
com.teamlava.bakerystory46	Bakery Story: Yoga Cafe	Storm8 Studios	Free	18,402	No	No	No	Yes	Yes	--
com.gummycrush.threezeroseventwo	1,2,3,6...3072	Gummy Crush, LLC	Free	18,209	No	No	No	No	No	--
com.igg.bravetrials	Brave Trials	IGG.COM	Free	18,142	No	No	No	Yes	Yes	--
com.and.games505.TerrariaPaid	Terraria.	505 Games Srl	Paid	15,321	Yes	No	No	Yes	No	--
com.tocaboca.hairsalon2	Toca Hair Salon 2	Toca Boca AB	Paid	14,855	Yes	No	Yes	Yes	Yes	--
com.tdr3.hs.android	HotSchedules	HotSchedules	Paid	14,078	Yes	No	No	Yes	No	--
com.trueaxis.trueskate	True Skate	True Axis	Paid	13,877	Yes	No	No	Yes	No	--
com.scottgames.fnaf2	Five Nights at Freddys 2	Scott Cawthon	Paid	13,362	Yes	No	No	Yes	No	--
com.coffeestainstudios.goatsimulator	Goat Simulator	Coffee Stain Studios	Paid	11,959	Yes	No	No	Yes	No	--
com.squareenixmontreal.hitmansniperandroid	Hitman: Sniper	SQUARE ENIX Ltd	Paid	11,479	Yes	No	No	Yes	No	--
com.ultimateguitar.tabs	Ultimate Guitar Tabs & Chords	Ultimate Guitar USA LLC	Paid	11,326	Yes	No	No	Yes	No	--
com.scottgames.fnaf3	Five Nights at Freddys 3	Scott Cawthon	Paid	10,706	Yes	No	No	Yes	No	--
com.rockstargames.gtasa	Grand Theft Auto: San Andreas	Rockstar Games	Paid	10,293	Yes	No	Yes	Yes	Yes	--
com.turner.cardwars	Card Wars - Adventure Time	Cartoon Network	Paid	9,986	Yes	No	No	Yes	No	--
com.teslacoilsw.launcher.prime	Nova Launcher Prime	TeslaCoil Software	Paid	9,641	No	No	No	No	No	Yes
com.square_enix.android_googleplay.FFT_en2	FINAL FANTASY TACTICS : WotL	SQUARE ENIX Co.,Ltd.	Paid	9,312	Yes	No	No	Yes	No	--

App ID (a)	App Name (b)	Developer Name (c)	Type (d)	Monthly App Downloads (e)	App is Dual-Homed (f)	Google App (g)	C++ App (h)	Developer is Dual-Homed (i)	Developer Wrote C++ Apps (j)	Microsft Counterpart Available (k)
com.FireproofStudios.TheRoom	The Room	Fireproof Games	Paid	9,280	Yes	No	No	Yes	No	--
com.eamobile.monopoly_na_wf	MONOPOLY	Electronic Arts Inc	Paid	8,581	Yes	No	No	Yes	Yes	--
com.mtvn.SBminigame	SpongeBobs Game Frenzy	Nickelodeon	Paid	8,509	Yes	No	No	Yes	No	--
org.prowl.torque	Torque Pro (OBD 2 & Car)	Ian Hawkins	Paid	8,453	No	No	No	No	No	No
com.ustwo.monumentvalley	Monument Valley	ustwo	Paid	7,824	Yes	No	No	Yes	No	--
com.edu.schollyapp	Scholly: Scholarship Search	Scholly	Paid	7,420	Yes	No	No	Yes	No	--
com.fsdigital.skinstudio	Minecraft Skin Studio	57Digital Ltd	Paid	7,346	Yes	No	No	Yes	No	--
com.paulart.go.launcherex.theme.pipboy3000	Nuclear Fallout 3k Multi Theme	Paul.ART	Paid	7,325	No	No	No	No	No	--
com.samruston.weather	Weather Timeline - Forecast	Sam Ruston	Paid	7,185	No	No	No	No	No	--
com.rockstargames.gtavc	Grand Theft Auto: Vice City	Rockstar Games	Paid	7,015	Yes	No	Yes	Yes	Yes	--
com.eamobile.nbajam_na_wf	NBA JAM by EA SPORTS™	ELECTRONIC ARTS	Paid	6,596	Yes	No	No	Yes	Yes	--
com.northcube.sleepcycle	Sleep Cycle alarm clock	Northcube AB	Paid	6,471	Yes	No	No	Yes	No	--
net.dinglisch.android.taskerm	Tasker	Crafty Apps EU	Paid	6,417	No	No	No	No	No	No
com.foxfi.key	FoxFi Key (supports PdaNet)	FoxFi Service	Paid	6,314	No	No	No	No	No	No
kb.Blek	Blek	kunabi brother	Paid	6,275	Yes	No	No	Yes	No	--
slide.watchFrenzy.premium	WatchMaker Premium Watch Face	androidslide	Paid	6,216	No	No	No	No	No	--
com.russellsoftworks.pipboywallpaper	OLD PipBoy 3000 Live Wallpaper	kopirat	Paid	6,118	No	No	No	No	No	--
com.lightricks.facetune	Facetune	Lightricks Ltd.	Paid	6,114	Yes	No	No	Yes	No	--
com.threeminutegames.lifeline.google	Lifeline	3 Minute Games, LLC	Paid	6,016	Yes	No	No	Yes	No	--
com.giantssoftware.fs14	Farming Simulator 14	GIANTS Software	Paid	5,961	Yes	No	Yes	Yes	Yes	--
com.maxmpz.audioplayer.unlock	Poweramp Full Version Unlocker	Max MP	Paid	5,924	No	No	Yes	No	Yes	--
com.dudeperfect.dudeperfect	Dude Perfect	Dude Perfect	Paid	5,890	Yes	No	No	Yes	No	--
com.turner.stevenrpg	Attack the Light	Cartoon Network	Paid	5,825	Yes	No	No	Yes	No	--
com.nuance.swype.dtc	Swype Keyboard	Nuance Communications, Inc	Paid	5,772	Yes	No	No	Yes	No	--
com.candyrufusgames.survivalcraft	Survivalcraft	Candy Rufus Games	Paid	5,688	Yes	No	Yes	Yes	Yes	--
com.gamesoulstudio.backflipmadness	Backflip Madness	Gamesoul Studio	Paid	5,651	Yes	No	Yes	Yes	Yes	--
com.epsxe.ePSXe	ePSXe for Android	epsxe software s.l.	Paid	5,556	No	No	No	No	No	--
com.weathersphere.noaa_hidef_radar	NOAA Hi-Def Radar	WeatherSphere	Paid	5,282	Yes	No	No	Yes	No	--
com.koushikdutta.cast.license	AllCast Premium	ClockworkMod	Paid	5,071	No	No	No	No	No	No
com.coffeestainstudios.goatsimulator.goatz	Goat Simulator GoatZ	Coffee Stain Studios	Paid	5,035	Yes	No	No	Yes	No	--
com.jeremysteckling.facerrel	Facer - Watch Faces	Little Labs, Inc.	Paid	4,975	No	No	No	Yes	No	--
com.etermax.preguntados.pro	Trivia Crack (Ad free)	Etermax	Paid	4,866	Yes	No	No	Yes	No	--
com.halfbrick.ageofzombies	Age of Zombies	Halfbrick Studios	Paid	4,835	Yes	No	No	Yes	Yes	--
air.com.flipline.papasfreezeriatogo	Papas Freezeria To Go!	Flipline Studios	Paid	4,827	Yes	No	No	Yes	No	--
com.atv.blackops	Call of Duty Black Ops Zombies	Activision Publishing, Inc.	Paid	4,763	Yes	No	No	Yes	Yes	--
com.ea.monopolymillionaire_na	MONOPOLY Millionaire	Electronic Arts Inc	Paid	4,696	Yes	No	No	Yes	Yes	--
com.popcap.pvz_na	Plants vs. Zombies™	ELECTRONIC ARTS	Paid	4,650	Yes	No	Yes	Yes	Yes	--
com.teamspeak.ts3client	TeamSpeak 3	TeamSpeak Systems GmbH	Paid	4,607	No	No	No	No	No	--
com.eamobile.sims3_na_qwf	The Sims 3	ELECTRONIC ARTS	Paid	4,578	Yes	No	Yes	Yes	Yes	--
com.aspyr.swkotor	Star Wars™: KOTOR	Aspyr Media, Inc.	Paid	4,501	Yes	No	No	Yes	No	--
com.mtvn.sbmigoogleplay	SpongeBob Moves In	Nickelodeon	Paid	4,279	Yes	No	No	Yes	No	--
com.fueled.afterlight	Afterlight	Afterlight Collective, Inc	Paid	4,256	Yes	No	No	Yes	No	--
com.active.aps.c25k	Couch to 5K®	ACTIVE Network, LLC	Paid	4,122	Yes	No	No	Yes	No	--
mahmed.net.synctuneswirelesspro	Sync iTunes to android - Pro	Bltstudlo	Paid	4,061	No	No	No	No	No	--
com.sonypicturestelevision.wheeloffortune30	Wheel of Fortune	Sony Pictures Television	Paid	4,041	Yes	No	No	Yes	No	--
com.gravitysensation.sumotori	Sumotori Dreams	gravitysensation	Paid	3,955	Yes	No	No	Yes	No	--
com.ngmoco.pocketgod	Pocket God™	Bolt Creative, Inc	Paid	3,930	Yes	No	No	Yes	No	--
com.nekki.vector.paid	Vector Full	NEKKI	Paid	3,887	Yes	No	No	Yes	No	--

App ID (a)	App Name (b)	Developer Name (c)	Type (d)	Monthly App Downloads (e)	App is Dual-Homed (f)	Google App (g)	C++ App (h)	Developer is Dual-Homed (i)	Developer Wrote C++ Apps (j)	Microsft Counterpart Available (k)
com.zenstudios.PortalPinball	Portal ® Pinball	Zen Studios	Paid	3,882	No	No	No	Yes	No	--
air.com.puffballsunited.stealingthediamond	Stealing the Diamond	PuffballsUnited	Paid	3,872	No	No	No	Yes	No	--
com.FireproofStudios.TheRoom2	The Room Two	Fireproof Games	Paid	3,817	Yes	No	No	Yes	No	--
com.ea.games.nfs13_na	Need for Speed Most Wanted	ELECTRONIC ARTS	Paid	3,774	Yes	No	No	Yes	Yes	--
com.wolfram.android.alpha	WolframAlpha	Wolfram Group	Paid	3,722	Yes	No	Yes	Yes	Yes	--
com.eightyeightgames.tenmillion	10000000	EightyEight Games	Paid	3,710	Yes	No	No	Yes	No	--
com.wdtinc.android.stormshield	Storm Shield	The E.W. Scripps Company	Paid	3,592	Yes	No	No	Yes	No	--
com.littleinferno.google	Little Inferno	Tomorrow Corporation	Paid	3,538	Yes	No	No	Yes	No	--
com.halfbrick.fruitninja	Fruit Ninja	Halfbrick Studios	Paid	3,530	Yes	No	Yes	Yes	Yes	--
com.flightradar24pro	Flightradar24 - Flight Tracker	Flightradar24 AB	Paid	3,518	Yes	No	No	Yes	No	--
com.headupgames.bridgeconstructor	Bridge Constructor	ClockStone STUDIO	Paid	3,517	Yes	No	No	Yes	No	--
com.loadcomplete.deadeyes	DEAD EYES	LoadComplete	Paid	3,506	Yes	No	No	Yes	No	--
com.keramidas.TitaniumBackupPro	Titanium Backup PRO Key ★ root	Titanium Track	Paid	3,454	No	No	No	No	No	No
com.squareenixmontreal.hitmango	Hitman GO	SQUARE ENIX Ltd	Paid	3,415	Yes	No	No	Yes	No	--
com.scannerradio_pro	Scanner Radio Pro	GordonEdwards.net LLC	Paid	3,398	No	No	No	No	No	--
eu.thedarken.sdm.unlocker	SD Maid Pro - Unlocker	darken	Paid	3,382	No	No	No	No	No	--
com.accuweather.paid.android	AccuWeather Platinum	Accuweather.com	Paid	3,338	Yes	No	No	Yes	No	--
com.magicnotion.kpmatchmaker	Kitty Powers Matchmaker	Magic Notion Ltd.	Paid	3,331	Yes	No	No	Yes	No	--
com.inxile.BardTale	The Bards Tale	inXile entertainment	Paid	3,262	Yes	No	No	Yes	No	--
com.jrtstudio.iSyncr	iSyncr for iTunes	JRT Studio	Paid	3,261	No	No	No	No	No	--
com.digidust.elokence.akinator.paid	Akinator the Genie	Elokence	Paid	3,251	Yes	No	No	Yes	No	--
com.quvideo.xiaoying.pro	VivaVideo Pro: HD Video Editor	QuVideo Inc.	Paid	3,219	Yes	No	No	Yes	No	--
com.ironhidegames.android.kingdomrushorigins	Kingdom Rush Origins	Ironhide Game Studio	Paid	3,206	Yes	No	Yes	Yes	Yes	--
com.notdoppler.earntodie	Earn to Die	Not Doppler	Paid	3,197	Yes	No	Yes	Yes	Yes	--
air.Airship2	Infiltrating the Airship	PuffballsUnited	Paid	3,132	No	No	No	Yes	No	--
net.kairosoft.android.gamedev3en	Game Dev Story	Kairosoft Co.,Ltd	Paid	3,114	No	No	No	Yes	No	--
com.syntellia.fleksy.keyboard	Fleksy + GIF Keyboard	Fleksy	Paid	3,110	Yes	No	No	Yes	No	--
com.playdead.limbo.full	LIMBO	Playdead	Paid	3,072	Yes	No	No	Yes	No	--
com.t2ksports.wwe2k15mobile	WWE 2K	2K Games, Inc.	Paid	3,028	Yes	No	No	Yes	No	--
com.rockstar.gta3	Grand Theft Auto III	Rockstar Games	Paid	2,991	Yes	No	Yes	Yes	Yes	--
com.rac7.DarkEcho	Dark Echo	RAC7	Paid	1,910	Yes	No	No	Yes	No	--
com.mozglabs.citysiedge	Zombie City Defense	Mozg Labs	Paid	1,883	No	No	No	No	No	--
modernpenandpaper.com.a21daycompletefix	21 Day Complete	The Modern Pen and Paper Company, LLC	Paid	1,846	No	No	No	No	No	--
net.inxile.tiq	The Impossible Quiz!	inXile entertainment	Paid	1,411	Yes	No	No	Yes	No	--
com.tribeplay.pandapet	Dr. Panda & Totos Treehouse	Dr. Panda Ltd	Paid	1,409	Yes	No	No	Yes	No	--
com.oddworld.Stranger	Oddworld: Strangers Wrath	Oddworld Inhabitants Inc	Paid	1,355	Yes	No	No	Yes	No	--
com.maxelus.blackholelivewallpaper	Supermassive Black Hole	maxelus.net	Paid	1,352	No	No	No	No	No	--
com.nomadgames.talisman	Talisman	Nomad Games	Paid	1,339	Yes	No	No	Yes	No	--
com.urbandroid.sleep.full.key	Sleep as Android Unlock	Urbandroid Team	Paid	1,302	No	No	No	No	No	--
com.bossastudios.ss13touch	Surgeon Simulator	Bossa Studios Ltd	Paid	1,238	Yes	No	No	Yes	No	--
com.tocaboca.tocalab	Toca Lab	Toca Boca AB	Paid	1,237	Yes	No	No	Yes	Yes	--
com.webprancer.google.garfieldescape	Garfields Escape Premium	Web Prancer	Paid	1,227	No	No	No	Yes	No	--
org.zooper.zwpro	Zooper Widget Pro	MYCOLORSCREEN	Paid	1,186	No	No	No	No	No	--
com.wb.lego.marvel	LEGO ® Marvel Super Heroes	Warner Bros. International Enterprises	Paid	1,174	No	No	No	Yes	Yes	--
com.foursakenmedia.heroesandcastles2	Heroes and Castles 2	Foursaken Media	Paid	1,149	Yes	No	No	Yes	No	--

App ID (a)	App Name (b)	Developer Name (c)	Type (d)	Monthly App Downloads (e)	App is Dual-Homed (f)	Google App (g)	C++ App (h)	Developer is Dual-Homed (i)	Developer Wrote C++ Apps (j)	Microsft Counterpart Available (k)
org.twisevictory.apps	Baby Wonder Weeks Milestones	Domus Technica	Paid	1,061	Yes	No	No	Yes	No	--
com.activision.gw3.dimensions	Geometry Wars 3: Dimensions	Activision Publishing, Inc.	Paid	1,006	Yes	No	No	Yes	Yes	--
com.rovio.angrybirdsstarwarsii.premium	Angry Birds Star Wars II	Rovio Entertainment Ltd.	Paid	995	Yes	No	Yes	Yes	Yes	--
com.argaming.myunturned	My Unturned	AR Gaming	Paid	983	No	No	No	No	No	--
com.mapmywalkplus.android2	Map My Walk+ GPS Pedometer	MapMyFitness, Inc.	Paid	909	Yes	No	No	Yes	No	--
com.glutenfreegames.catsimulator	Stray Cat Simulator	Gluten Free Games	Paid	905	Yes	No	No	Yes	No	--
ch.threema.app	Threema	Threema GmbH	Paid	866	Yes	No	No	Yes	No	--
vo.threes.exclaim	Threes!	Sirvo llc	Paid	774	Yes	No	No	Yes	No	--
com.paradoxplaza.kopp2	Knights of Pen & Paper 2	Paradox Interactive	Paid	740	Yes	No	No	Yes	No	--
com.flavionet.android.camera.pro	Camera FV-5	FGAE	Paid	704	No	No	No	No	No	--
com.onyxdev.iconpack.theme.glif	Glif - Icon Pack	Prodigy Arts, Inc	Paid	659	No	No	No	No	No	--
com.mapmyrunplus.android2	Run with Map My Run +	MapMyFitness, Inc.	Paid	654	Yes	No	No	Yes	No	--
com.flukedude.impossiblegame	The Impossible Game	FlukeDude	Paid	652	Yes	No	Yes	Yes	Yes	--
com.apalon.myclock	My Alarm Clock	Apalon Apps	Paid	647	Yes	No	No	Yes	No	--
com.vertumus.rewun	Rewun - Icon Pack	Vertumus	Paid	611	No	No	No	No	No	--
com.webprancer.google.homesweetgarfield	Home Sweet Garfield Live WP	Web Prancer	Paid	580	No	No	No	Yes	No	--
com.paulart.golocker.pipboy3000	Go Locker Nuclear Fallout 3k	Paul.ART	Paid	529	No	No	No	No	No	--
com.jb.gokeyboard.theme.paulart.pipboy3000	Go Keyboard Nuclear Fallout 3k	Paul.ART	Paid	523	No	No	No	No	No	--
com.mageeks.android.farmingpro2015	Farming PRO 2015	Mageeks Apps & Games	Paid	522	Yes	No	No	Yes	No	--
com.dianxinos.dxbs.paid	DU Battery Saver PRO & Widgets	DU APPS STUDIO	Paid	511	No	No	No	No	No	--
com.turner.mixelrush	Mixels Rush	Cartoon Network	Paid	473	Yes	No	No	Yes	No	--
joybits.doodlegod	Doodle God™	JoyBits Co. Ltd.	Paid	441	Yes	No	No	Yes	No	--
com.disney.castleofillusion_goo	Castle of Illusion	Disney	Paid	416	Yes	No	No	Yes	Yes	--
com.jundroo.simplerockets	SimpleRockets	Jundroo, LLC	Paid	409	Yes	No	Yes	Yes	Yes	--
ginlemon.flowerpro	Smart Launcher Pro 3	GinLemon	Paid	408	No	No	No	No	No	--
com.tt2kgames.xcomew	XCOM®: Enemy Within	2K Games, Inc.	Paid	353	Yes	No	No	Yes	No	--
io.turnerdavis.pipdroid	PipDroid for UCCW (Paid)	Turner Davis	Paid	333	No	No	No	No	No	--
com.wb.lego.tcs	LEGO® Star Wars™: TCS	Warner Bros. International Enterprises	Paid	327	Yes	No	No	Yes	Yes	--
com.jumpgames.RealSteel	Real Steel	Reliance Big Entertainment (UK) Private Limited	Paid	323	Yes	No	Yes	Yes	Yes	--
com.beamdog.baldursgateenhancededition	Baldurs Gate Enhanced Edition	Beamdog	Paid	239	Yes	No	No	Yes	No	--
com.stmp.minimalface	Watch Face - Minimal & Elegant	Studio eXtreme	Paid	230	No	No	No	No	No	--
alexcrusher.just6weeks.full	Just 6 Weeks	Just Do Inc.	Paid	202	No	No	No	No	No	--
com.gameloft.android.ANMP.GloftASHM	The Amazing Spider-Man 2	Gameloft	Paid	198	Yes	No	Yes	Yes	Yes	--
com.astragon.cs2014	Construction Simulator 2014	astragon Entertainment GmbH	Paid	196	Yes	No	No	Yes	No	--
com.turner.fcallstars	Formula Cartoon All Stars	Cartoon Network	Paid	121	Yes	No	No	Yes	No	--
stericson.busybox.donate	BusyBox Pro	Stephen (Stericson)	Paid	120	No	No	No	No	No	--
com.mapmyrideplus.android2	Map My Ride+ GPS Cycling	MapMyFitness, Inc.	Paid	109	Yes	No	No	Yes	No	--
com.the3d4medical.EssentialAnatomy	Essential Anatomy 3	3D4Medical.com, LLC	Paid	104	Yes	No	No	Yes	No	--
com.realdrift.sipon	Real Drift Car Racing	RealGames	Paid	94	No	No	No	No	No	--
com.rt.hook	Hook	Rainbow Train	Paid	93	Yes	No	No	Yes	No	--

Notes: Top 100 downloaded apps are collected using daily top 100 downloaded free and paid apps in June 2015.

An app is dual-homed if it was launched on both iOS and Android platforms.

A developer is dual-homed if the developer has published top downloaded apps on both iOS and Android platforms.

C++ app is defined as an app written in C++.

Microsoft counterpart is identified for top downloaded apps that are neither google apps, apps with dual-homed developers or apps whose developers have written C++ apps.

Programming languages are identified by Google, and from Top Android Applications Languages in the Export Report of Dr. Cox

Sources: "AppAnnie Daily DNA Data," App Annie, January 26, 2016.

Expert Report of Dr. Cox, Exhibit 1b and Exhibit 1c, Octorber 3, 2011.

Exhibit 3a
Project Cost

2008

Salaries for Two Developers	
Salary	$ 231,460
Benefits	107,427
Total Salary	$ 338,887
Time Devoted to Project	3 months
Cost of Project	**$ 84,722**

Notes: Salary for an applications software developer in the San Jose-Sunnyvale-Santa Clara counties of California.

Benefits account for 31.7% of total compensation for the information sector in March 2008.

Sources: "Occupational Employment and Wages: 15-1032 Computer Software Engineers, Systems Software," Bureau of Labor Statistics, May 2008, http://www.bls.gov/oes/2008/may/oes151032.htm.

"Employer Costs for Employee Compensation: Historical Listing," Bureau of Labor Statistics, March 2004 - September 2015, http://www.bls.gov/ncs/ect/sp/ececqrtn.txt.

Exhibit 3b
Avoided Cost of Developing Android Applications

2008 - 2015

Cost of Developing Android Applications - High End		
Number of Applications to be Developed		1,000
Cost of Development per Application	$	100,000
Avoided Cost of App. Development - High End	**$**	**100,000,000**
Cost of Developing Android Applications - Low End		
Number of Applications to be Developed		1,000
Cost of Development per Application - Low End	$	22,673
Avoided Cost of App. Development - Low End	**$**	**22,673,000**

Notes: Number of applications to be developed is based on monthly top 200 Apps by usage provided by comScore Inc., and is calculated based on econometric analysis.

Multi-homed applications, and applications written in C++ or developed by Google have been eliminated from the analysis.

Sources: Deposition of Reto Meier, December 11, 2015.

"Mobile Metrix," comScore Inc., January 19, 2016.

"AppAnnie Daily DNA Data," App Annie, January 26, 2016.

http://www.statista.com/statistics/256541/average-cost-to-develop-an-app-by-os/.

Exhibit 3c
Avoided Cost of C++ Training

2008 - 2015

(1)	Number of Developers with an Application Appearing in a Top 100 Daily List , 2012-2015 [1]		1,889
(2)	Number of Developers to Receive C++ Training, 2012-2015 [2]		986
(3)=2*(2)	Number of Developers to Receive C++ Training, 2008-2015 [3]		1,972
(4)=1.6*(3)	Number of Programmers to Receive C++ Training, 2008-2015 [4]		3,155
(5)	Cost of C++ Training Per Programmer	$	715
(6)=(5)*(4)	Total Avoided Cost of C++ Training, 2008-2015	**$**	**2,255,968**

Notes: [1] Conservatively calculated based on App Annie daily data for top 100 applications on Google Play from 2012-2015.

[2] Calculated as a number of non-Google developers with applications written in a language other than C++, and that are not multi-homed. This is based on information received from Google and Dr. Cox Exhibits 1b and 1c that identified applications written in C++.

[3] The number of developers to receive C++ training for 2012-2015 is then multiplied by two to reflect the number of developers to receive C++ training for 2008-2015.

[4] Calculated as the average number of programmers per developer times the number of developers to receive C++ training in 2008-2015. The average number of programmers per developer is 1.6 and is calculated based on the average number of programmers per developer calculated from Android's Developer Challenge 2008.

Sources: "AppAnnie Daily DNA Data," App Annie, January 26, 2016.

Expert Report of Dr. Cox, Exhibits 1b and 1c, October 3, 2011.

http://extension.berkeley.edu/search/publicCourseSearchDetails.do?method=load&courseId=40931.

http://extension.ucsd.edu/studyarea/index.cfm?vCourse=CSE-40475.

"Google Reveal The Top 50 Android Applications (46 Public)," Chris Moor at Talk Android, http://www.talkandroid.com/92-developer-challenge-top-50-android-application/.

"Splash Play," AndroidTapp, http://www.androidtapp.com/splashplay/.

Exhibit 3d.1
Share Loss Analysis
Jan. 2008 - Dec. 2015

	2008 (a)	2009 (b)	2010 (c)	2011 (d)	2012 (e)	2013 (f)	2014 (g)	2015 (h)	Total (i)
Revenue (Share Loss)									
Ads	$ 0.0	$ 0.3	$ 2.6	$ 12.4	$ 46.8				
Hardware	--	--	2.3	0.0	6.0				
Apps	0.0	0.0	0.2	0.7	2.7				
Digital Content	--	--	--	0.3	2.1				
Total	**$ 0.0**	**$ 0.4**	**$ 5.0**	**$ 13.4**	**$ 57.6**				
Cost of Sales (Share Loss)									
TAC	$ 0.0	$ 0.1	$ 0.9	$ 2.4	$ 12.0				
Hardware	--	--	--	0.0	6.7				
Apps	--	--	--	0.0	1.2				
Digital Content	--	--	--	0.5	3.3				
Infrastructure & Other COS	--	--	--	1.3	1.9				
Operations	0.0	0.0	0.1	--	--				
COS (including DTC)	0.0	0.0	2.2	--	--				
Total	**$ 0.0**	**$ 0.1**	**$ 3.1**	**$ 4.2**	**$ 25.1**				
Gross Profit									
Total Gross Profit	$ 0.0	$ 0.3	$ 1.9	$ 9.2	$ 32.4				
Gross Margin (%)	44.1 %	78.3 %	37.6 %	69.0 %	56.3 %				%
Operating Expenses (Share Loss)									
Android Engineering PM	$ --	$ --	$ --	$ --	$ --				
Android Marketing	--	--	--	--	--				
Android Legal	--	--	--	--	--				
Android Sales and Other	--	--	--	--	--				
Android General and Administrative	--	--	--	--	--				
Incremental Search and Advertising Expenses	0.0	0.0	0.2	1.0	3.9				
Total	**$ 0.0**	**$ 0.0**	**$ 0.2**	**$ 1.0**	**$ 3.9**				
Android Advertising Share Loss	2.2 %	2.2 %	2.2 %	2.2 %	2.2 %				
Google Play Share Loss	2.0	2.0	2.0	2.0	2.0				
Diversion Ratio	[redacted]								
Search Share	100.0	76.0	67.3	76.9	67.1				
Gross Loss of Profit	$ 0.0	$ 0.3	$ 1.7	$ 8.2	$ 28.5				
iPhone Offset	[redacted]								
Net Loss of Profit	**$**								

Notes: In millions of U.S. dollars.

Ads revenue, TAC and incremental search and advertising expenses are from Exhibit 1a.1 and multiplied by the Android share loss. Revenue and COS for hardware, apps and digital content are from Exhibit 1a.1 and multiplied by the Google Play share loss. For the share loss refer to Exhibit 3d.3 under Scenario 1, for the 2008 share loss, 2009 is used.

The gross loss of profit is calculated as (total revenue - total COS - incremental search and advertising expenses).

The iPhone offset calculated as (diversion ratio)*[(Revenue-TAC)*(1-search share*[redacted])-incremental search and operating expenses].

For the diversion ratio refer to Exhibit 3d.2. For the 2008 diversion ratio, 2009 is used.

For the search share refer to search advertising as a share of Android advertising revenue from Exhibit 1c.

The annual margin impact figures of [redacted] and [redacted] are from GOOG-00130338 at 343.

Net loss of profit is the gross loss of profit with the added iPhone offset.

Sources: Exhibit 1a.1.

Exhibit 1c.

Exhibit 3d.2.

GOOG-00130338.

Exhibit 3d.2
Overall Revenue Loss Percentages and Diversion Ratios

2009 - 2015

	2009 (a)	2010 (b)	2011 (c)	2012 (d)	2013 (e)	2014 (f)	2015 (g)
Percentage of Revenue in the U.S.							
Percentage of Ad Revenue in the US				45.6 %	[redacted]		
Percentage of Play Revenue in the US				23.2			
Revenue Loss and Diversion Ratio							
Ad Revenue Loss	-2.2 %	-2.2 %	-2.2 %	-2.2 %			
Ad Revenue Diversion Ratio (from Android to iOS)	[redacted]						
Play Revenue Loss	-2.0	-2.0	-2.0	-2.0			

Notes: Revenue loss and diversion ratios for 2009, 2010, and 2011 are set equal to 2012 values.
U.S. revenue percentages are set equal to 2013 value.

Sources: Exhibit 3d.3
Exhibit 3d.4.
Exhibit 3d.5.

Exhibit 3d.3
User Loss and Diversion Ratios For U.S. and Rest of World

2012-2015

	2012 (a)	2013 (b)	2014 (c)	2015 (d)
United States				
User Loss	[redacted]			
Diversion Ratio (Android to iOS)				
Rest of World				
User Loss				
Diversion Ratio (Android to iOS)				
Worldwide				
Diversion Ratio (Android to iOS)				

Sources: Calculations.

Exhibit 3d.4
U.S. Revenue as a Percentage of Worldwide Revenue For Ad and Play

	Phone	Tablet	Tablet Share of Devices	Overall US Share	2013 Revenue	Share of the Total
	(a)	(b)	(c)	(d)	(e)	(f)

U.S. Revenue as % of Worldwide Revenue By Device Type

Play Revenue
Play

Ads Revenue
Search Ads
Display Ads
Total Ads

[redacted]

Notes: For U.S. revenue as a % of worldwide revenue for Play, Search Ad, and Display Ad refer to GOOG-00186889.

For tablet share of devices refer to GOOG-00186879.

For 2013 revenue for search ads and display ads refer to Exhibit 1c. Display ads includes AdSense.

Sources: Exhibit 1c.

GOOG-00186889.

GOOG-00186879.

Exhibit 3d.5
Android Shipments for the U.S. and Worldwide

2008 Q4 - 2015 Q3

	2008 (a)	2009 (b)	2010 (c)	2011 (d)	2012 (e)	2013 (f)	2014 (g)	2015 Q1-Q3 (h)
United States	661.3	4,915.2	28,778.6	57,211.8	66,601.2	75,364.0	96,229.0	68,617.5
Worldwide	691.4	7,011.6	71,101.6	243,422.8	501,424.7	802,274.5	1,060,470.1	844,264.9
U.S. as a Share of Worldwide Units	95.6 %	70.1 %	40.5 %	23.5 %	13.3 %	9.4 %	9.1 %	8.1 %

Notes: In millions of units.
Android shipment data is available from Q4 2008 to Q3 2015.

Source: "WW Quarterly Mobile Phone Tracker," IDC, November 13, 2015.

Exhibit 3e
Top Down Apportionment

Number of Lines of Implementation Code	
Implementation Code for the 37 APIs	259,474
Total Android Source Code	15,347,169
Share of the Total	1.7 %

Number of Lines of Implementation Code	
Implementation Code for the 37 APIs	259,474
Total Android, Search and Ads Code	65,547,169
Share of the Total	0.4 %

Android Profit	[redacted]
Android-Related Profit	
Android Profit (Apportioned to the 37 API Packages)	
Android-Related Profit (Apportioned to the 37 API Packages)	

Notes: The implementation code is based on Android's Gingerbread release. See the Expert Report of Dr. Astrachan, p. 55.

The Search and Ads code is based on a conversation with Mr. Ghuloum, Engineering Director for Android. The Search and Ads code includes 50.2 million lines of code.

Android profit from Exhibit 1a.4.

Android-related profit from Exhibit 1a.1.

Android profit (apportioned to the 37 API packages) is the product of Android profit and 1.7%.

Android-related profit (apportioned to the 37 API packages) is the product of Android-related profit and 0.4%.

Sources: Exhibit 1a.4.

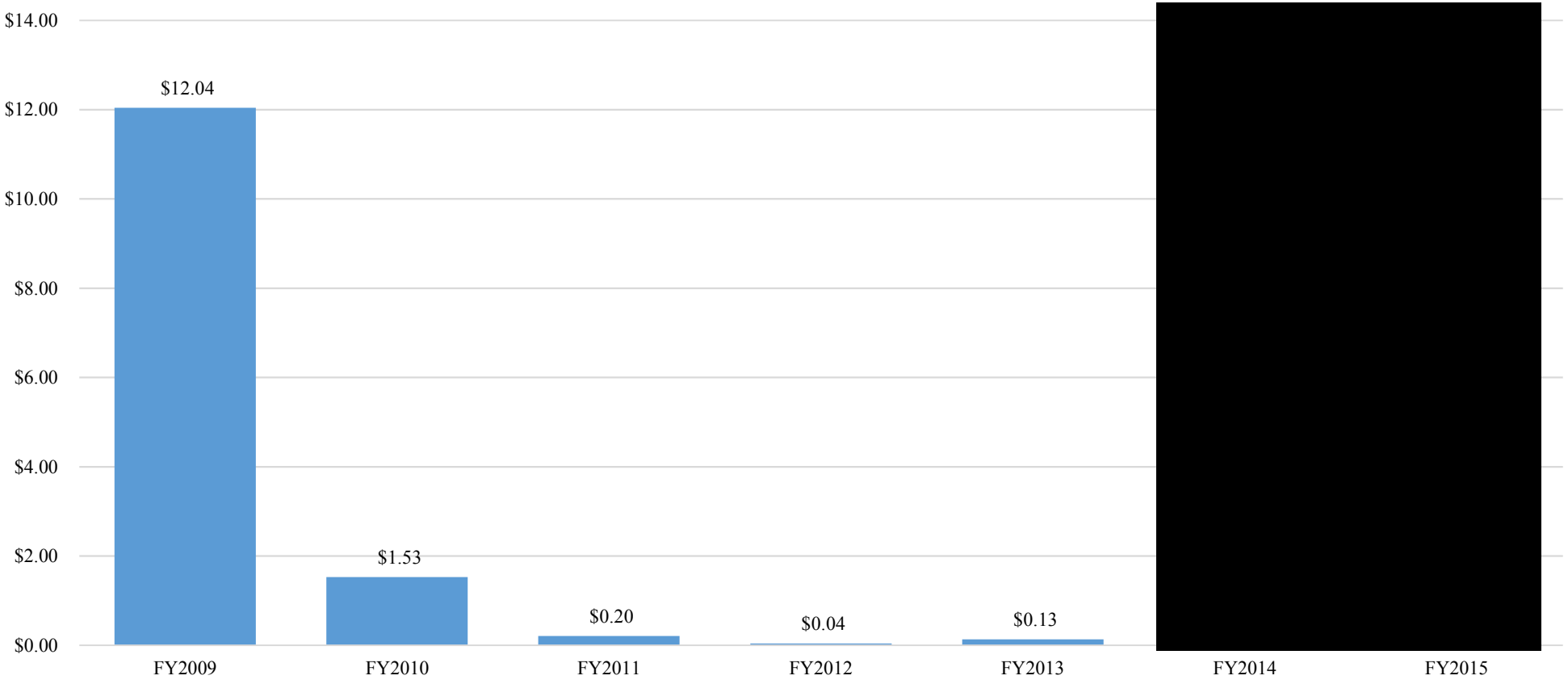
Exhibit 1a.1.

Expert Report of Dr. Astrachan, January 8, 2016.

Conversation with Mr. Ghuloum, Engineering Director for Android

Exhibit 4a
Mr. Malackowski's Claimed Lost Profits Per Android Device

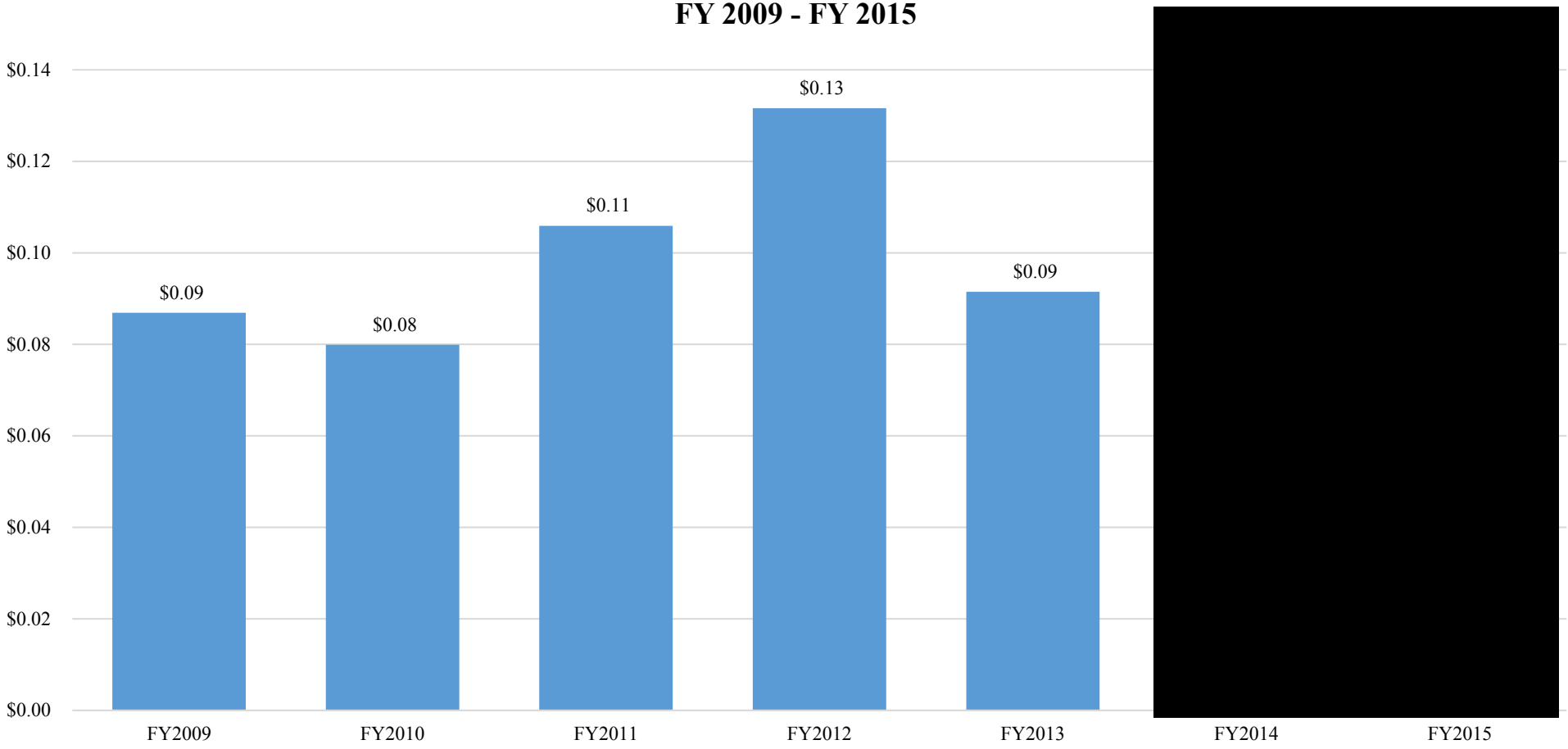
FY 2009 - FY 2015



Notes: Lost profits per Android device are calculated using Mr. Malackowski's Java ME damages and actual Android volumes.
Android volumes are adjusted to reflect Sun's and Oracle's fiscal years.

Sources: Expert Report of James E. Malackowski, January 8, 2016, Exhibit 12.
"WW Quarterly Mobile Phone Tracker - 2015Q3 Historical Release," IDC, 2015.

Exhibit 4b
Java ME Licensing Revenue Per Feature Phone
FY 2009 - FY 2015



Notes: Handset volumes are adjusted to reflect Sun's and Oracle's fiscal years.

Sources: Expert Report of James E. Malackowski, January 8, 2016, Exhibit 12.
"WW Quarterly Mobile Phone Tracker - 2015Q3 Historical Release," IDC, 2015.
OAGOOGLE0000702509.
OAGOOGLE2000003713.

Exhibit 4c
Java ME Lost Profits
Using Mr. Malackowski's Approach, But Substituting the "Best Estimate" Forecast - Excluding Java TV
FY 2009 - FY 2015

		FY2009 (a)	FY2010 (b)	FY2011 (c)	FY2012 (d)	FY2013 (e)	FY2014 (f)	FY2015 (g)	Total (h)
(1)	Java ME Forecasted Licensing Revenue[1]	$ 75,949,318	$ 74,508,713	$ 73,817,562	$ 69,361,215	$ 65,173,897	[redacted]		$ 477,592,431
(2)	Java ME Licensing Revenue[2]	$ 96,951,229	$ 100,657,682	$ 123,610,000	$ 150,198,000	$ 86,754,824			$ 608,568,092
(3) = (1)-(2)	*Lost Java ME Revenue*	$ (21,001,911)	$ (26,148,969)	$ (49,792,438)	$ (80,836,785)	$ (21,580,927)			$ (130,975,661)
(4)	Incremental Expense as a % of Lost Revenue[3]	17.6 %	17.6 %	10.0 %	9.8 %	9.5 %			
(5) = (3)*(4)	*Incremental Expenses*	$ (3,692,740)	$ (4,597,741)	$ (4,987,531)	$ (7,924,046)	$ (2,050,953)			$ (12,110,826)
(6) = (3)-(5)	**Adjusted Java ME Lost Profits**	**$ (17,309,171)**	**$ (21,551,229)**	**$ (44,804,907)**	**$ (72,912,739)**	**$ (19,529,974)**			**$ (118,864,835)**

Notes: FY 2009 revenue data reflects Sun's fiscal year ending in June; FY 2010 - FY 2015 data reflect Oracle's fiscal years ending in May.

[1] Java ME revenue forecasts for 2009-2012 are taken from the "Best Estimate" forecast in the October 21, 2008 Java in Wireless Business Review. See OAGOOGLE0000142142. Data for 2012-2015 are projected using the 2011-2012 growth rate from the forecast. The forecast excludes CDC-TV royalties.

[2] Java ME revenue data for 2009-2010 is taken from OAGOOGLE0000702509 "Mapping" tab. Revenue data for 2011-2015 is taken from OAGOOGLE2000003713 "Lic Revenue by Product" tab.

[3] Incremental expense data for 2009 is the sum of "Incremental COGS" and "Incremental Sales Expense" from Sun's 2006 Java ME P&L statement. See OAGOOGLE0005039944 at 946. Data for 2011-2015 are taken from Oracle's 2011-2015 Java Financials. See OAGOOGLE2000003713.

Sources: Expert Report of James E. Malackowski, January 8, 2016, Exhibits 12 - 12.10.
OAGOOGLE0000142142.
OAGOOGLE0000702509.
OAGOOGLE2000003713.
OAGOOGLE0005039944.

Exhibit 4d
Java ME Lost Profits
Using Mr. Malackowski's Approach, But Substituting the "Best Estimate" Forecast - Including Java TV
FY 2009 - FY 2015

		FY2009 (a)	FY2010 (b)	FY2011 (c)	FY2012 (d)	FY2013 (e)	FY2014 (f)	FY2015 (g)	Total (h)
(1)	Java ME Forecasted Licensing Revenue[1]	$ 86,306,043	$ 103,438,713	$ 113,317,562	$ 99,791,215	$ 87,879,464	[redacted]		$ 636,274,415
(2)	Java ME Licensing Revenue[2]	$ 96,951,229	$ 100,657,682	$ 123,610,000	$ 150,198,000	$ 86,754,824			$ 608,568,092
(3) = (1)-(2)	*Lost Java ME Revenue*	$ (10,645,186)	$ 2,781,031	$ (10,292,438)	$ (50,406,785)	$ 1,124,640			$ 27,706,323
(4)	Incremental Expense as a % of Lost Revenue[3]	17.6 %	17.6 %	10.0 %	9.8 %	9.5 %			29.4 %
(5) = (3)*(4)	*Incremental Expenses*	$ (1,871,730)	$ 488,985	$ (1,030,957)	$ (4,941,138)	$ 106,881			$ 8,148,964
(6) = (3)-(5)	**Adjusted Java ME Lost Profits**	**$ (8,773,456)**	**$ 2,292,045**	**$ (9,261,481)**	**$ (45,465,647)**	**$ 1,017,759**			**$ 19,557,359**

Notes: FY 2009 revenue data reflects Sun's fiscal year ending in June; FY 2010 - FY 2015 data reflect Oracle's fiscal years ending in May.

[1] Java ME revenue forecasts for 2009-2012 are taken from the "Best Estimate" forecast in the October 21, 2008 Java in Wireless Business Review. See OAGOOGLE0000142142. Data for 2012-2015 are projected using the 2011-2012 growth rate from the forecast. The forecast includes CDC-TV royalties.

[2] Java ME revenue data for 2009-2010 are taken from OAGOOGLE0000702509 "Mapping" tab. Revenue data for 2011-2015 are taken from OAGOOGLE2000003713 "Lic Revenue by Product" tab.

[3] Incremental expense data for 2009 is the sum of "Incremental COGS" and "Incremental Sales Expense" from Sun's 2006 Java ME P&L statement. See OAGOOGLE0005039944 at 946. Data for 2011-2015 are taken from Oracle's 2011-2015 Java Financials. See OAGOOGLE2000003713.

Sources: Expert Report of James E. Malackowski, January 8, 2016, Exhibits 12 - 12.10.
OAGOOGLE0000142142.
OAGOOGLE0000702509.
OAGOOGLE2000003713.
OAGOOGLE0005039944.

Exhibit 4e
Java ME Lost Profits
But-for Capture Calculation

FY 2009 - FY 2015

		FY2009 (a)	FY2010 (b)	FY2011 (c)	FY2012 (d)	FY2013 (e)	FY2014 (f)	FY2015 (g)	Total (h)
(1)	Java ME Licensing Revenue[1]	$ 96,951,229	$ 100,657,682	$ 123,610,000	$ 150,198,000	$ 86,754,824			$ 608,568,092
	Adjustment Based on Market Share								
(2)	Revenue Capture Rate[2]	0.2 %	1.3 %	8.5 %	24.3 %	54.3 %			
(3) = (1)*(2)	*Lost Java ME Licensing Revenue*	$ 146,314	$ 1,282,212	$ 10,468,381	$ 36,470,882	$ 47,147,577			$ 145,748,209
(4)	Incremental Expense as a % of Lost Revenue[3]	17.6 %	17.6 %	10.0 %	9.8 %	9.5 %			11.8 %
(5) = (3)*(4)	*Incremental Expenses*	$ 25,726	$ 225,450	$ 1,048,580	$ 3,575,068	$ 4,480,691			$ 17,232,031
(6) = (3)-(5)	**Java ME Lost Profits**	**$ 120,588**	**$ 1,056,763**	**$ 9,419,801**	**$ 32,895,815**	**$ 42,666,886**			**$ 128,516,178**
	Adjustment Based on Diversion Ratio								
(7)	Revenue Capture Rate[2]	0.1 %	0.7 %	5.0 %	14.8 %	36.5 %			
(8) = (1)*(7)	*Lost Java ME Licensing Revenue*	$ 83,123	$ 734,142	$ 6,122,220	$ 22,236,220	$ 31,665,223			$ 97,416,521
(9)	Incremental Expense as a % of Lost Revenue[3]	17.6 %	17.6 %	10.0 %	9.8 %	9.5 %			11.8 %
(10) = (8)*(9)	*Incremental Expenses*	$ 14,615	$ 129,083	$ 613,241	$ 2,179,711	$ 3,009,319			$ 11,687,247
(11) = (1)-(10)	**Java ME Lost Profits**	**$ 68,507**	**$ 605,059**	**$ 5,508,979**	**$ 20,056,509**	**$ 28,655,904**			**$ 85,729,274**

Notes: FY 2009 revenue data reflects Sun's fiscal year ending in June; FY 2010 - FY 2015 data reflect Oracle's fiscal years ending in May.

[1] Java ME revenue data for 2009-2010 are taken from OAGOOGLE0000702509 "Mapping" tab. Revenue data for 2011-2015 are taken from OAGOOGLE2000003713 "Lic Revenue by Product" tab.

[2] See Exhibit 4f.

[3] Incremental expense data for 2009 is the sum of "Incremental COGS" and "Incremental Sales Expense" from Sun's 2006 Java ME P&L statement. See OAGOOGLE0005039944 at 946. Data for 2011-2015 are taken from Oracle's 2011-2015 Java Financials. See OAGOOGLE2000003713.

Sources: OAGOOGLE0000702509.
OAGOOGLE2000003711.
OAGOOGLE0005039944.
OAGOOGLE2000003713.

Exhibit 4f
But-for Java ME Revenue Capture Rates

FY 2009 - FY 2015

		FY2009 (a)	FY2010 (b)	FY2011 (c)	FY2012 (d)	FY2013 (e)	FY2014 (f)	FY2015 (g)	Total (h)
(1)	Total Handset Units	1,276,577,524	1,455,988,268	1,653,544,456	1,719,591,082	1,799,024,686	1,885,107,543	1,962,186,444	11,752,020,003
(2)	Android Units	1,923,643	18,313,597	129,103,093	335,969,248	633,443,871	912,391,740	1,104,512,243	3,135,657,435
(3) = (1) - (2)	*Non-Android Handsets*	1,274,653,881	1,437,674,671	1,524,441,362	1,383,621,834	1,165,580,815	972,715,803	857,674,201	8,616,362,568
(4)	iPhone Units	18,784,872	32,188,110	65,400,237	113,372,547	141,184,951	162,843,234	218,333,678	752,107,628
(5) = (3) - (4)	*Potential Java ME Licensed Handsets*[1]	1,255,869,009	1,405,486,561	1,459,041,125	1,270,249,287	1,024,395,865	809,872,569	639,340,523	7,864,254,939
	Adjustment Based on Market Share								
(6) = (5)/(3)	Potential Java ME Market Share	98.5 %	97.8 %	95.7 %	91.8 %	87.9 %	83.3 %	74.5 %	
(7) = [(6)*(2)]/(5)	**But-for Java ME Revenue Capture Rate**	**0.2 %**	**1.3 %**	**8.5 %**	**24.3 %**	**54.3 %**	**93.8 %**	**128.8 %**	
	Adjustment Based on Diversion Ratio								
(8)	Diversion Ratio[2]	56.0 %	56.0 %	56.0 %	56.0 %	59.0 %	59.5 %	57.7 %	
(9) = [(8)*(2)]/(5)	**But-for Java ME Revenue Capture Rate**	**0.1 %**	**0.7 %**	**5.0 %**	**14.8 %**	**36.5 %**	**67.1 %**	**99.8 %**	

Notes: Volume figures are adjusted to reflect Oracle's June - May fiscal year

[1] Potential Java ME Licensed Handset volume is calculated as the sum of non-Android and non-iOS handsets.

[2] See Exhibit 3d.2.

Source: "WW Quarterly Mobile Phone Tracker - 2015Q3 Historical Release," IDC, 2015

Exhibit 5a
Worldwide Mobile Phone Shipment Shares by Operating System

2007 - 2015

	2007 (a)	2008 (b)	2009 (c)	2010 (d)	2011 (e)	2012 (f)	2013 (g)	2014 (h)	Q3 2015 (i)
Operating System									
Android	0.00 %	0.05 %	0.52 %	4.46 %	14.17 %	28.76 %	43.44 %	54.05 %	59.14 %
BlackBerry OS	1.07	1.79	2.57	3.06	2.97	1.86	1.04	0.29	0.20
Firefox OS	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.07	0.05
iOS	0.32	1.04	1.87	2.98	5.42	7.79	8.31	9.82	10.98
Linux	1.23	1.34	0.79	0.57	0.84	0.83	0.39	0.22	0.14
Maemo/MeeGo	0.00	0.00	0.01	0.04	0.06	0.04	0.00	0.00	0.00
Other	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Palm OS	0.18	0.20	0.03	0.00	0.00	0.00	0.00	0.00	0.00
RTOS	89.15	88.53	87.08	80.90	71.23	58.32	44.83	33.65	27.61
Sailfish OS	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Symbian	6.89	5.55	5.81	6.95	4.75	1.37	0.14	0.12	0.09
Tizen	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.09
webOS	0.00	0.00	0.12	0.11	0.05	0.00	0.00	0.00	0.00
Windows Mobile	1.15	1.49	1.20	0.84	0.18	0.02	0.00	0.00	0.00
Windows Phone	0.00	0.00	0.00	0.10	0.34	1.00	1.81	1.78	1.70
Grand Total	**100 %**	**100 %**	**100 %**	**100 %**	**100 %**	**100 %**	**100 %**	**100 %**	**100 %**

U.S. Mobile Phone Shipment Shares by Operating System

2007 - 2015

	2007 (j)	2008 (k)	2009 (l)	2010 (m)	2011 (n)	2012 (o)	2013 (p)	2014 (q)	Q3 2015 (r)
Android	0.00 %	0.38 %	2.73 %	15.71 %	29.98 %	38.12 %	43.45 %	51.95 %	51.12 %
BlackBerry OS	4.54	8.87	11.92	10.76	5.26	1.44	1.09	0.31	0.27
Firefox OS	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00
iOS	1.93	4.04	5.62	7.65	17.11	27.30	30.38	33.57	32.69
Linux	0.27	0.67	0.33	0.01	0.00	0.00	0.00	0.00	0.00
Maemo/MeeGo	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Palm OS	1.00	1.39	0.19	0.00	0.00	0.00	0.00	0.00	0.00
RTOS	88.95	80.64	74.55	61.92	44.93	31.37	23.12	12.13	13.92
Symbian	0.28	0.35	0.64	1.11	0.40	0.00	0.00	0.00	0.00
webOS	0.00	0.00	0.83	0.88	0.43	0.00	0.00	0.00	0.00
Windows Mobile	3.03	3.66	3.19	1.61	0.16	0.00	0.00	0.00	0.00
Windows Phone	0.00	0.00	0.00	0.34	1.72	1.77	1.95	2.04	1.99
Grand Total	**100 %**	**100 %**	**100 %**	**100 %**	**100 %**	**100 %**	**100 %**	**100 %**	**100 %**

Note: Data is available from 2007 to Q3 2015.

Source: *IDC*, WW Quarterly Mobile Phone Tracker, November 13, 2015.

Exhibit 5b
Mobile Phone Shipment Shares by Region

2007 - 2015

	2007 (a)	2008 (b)	2009 (c)	2010 (d)	2011 (e)	2012 (f)	2013 (g)	2014 (h)	Q3 2015 (i)
USA									
Feature Phone	88.95 %	80.64 %	74.55 %	61.92 %	44.93 %	31.37 %	23.12 %	12.13 %	13.92 %
Smartphone	11.05	19.36	25.45	38.08	55.07	68.63	76.88	87.87	86.08
Rest of the World									
Feature Phone	89.19 %	89.75 %	89.02 %	83.36 %	74.51 %	61.32 %	47.08 %	35.90 %	29.03 %
Smartphone	10.81	10.25	10.98	16.64	25.49	38.68	52.92	64.10	70.97
Worldwide									
Feature Phone	89.15 %	88.53 %	87.08 %	80.90 %	71.23 %	58.32 %	44.83 %	33.65 %	27.61 %
Smartphone	10.85	11.47	12.92	19.10	28.77	41.68	55.17	66.35	72.39

Note: Data is available from 2007 to Q3 2015.

Source: *IDC* , WW Quarterly Mobile Phone Tracker, November 13, 2015

Exhibit 5c
Worldwide Mobile Phone Shipments by Operating System

2007 - 2015

	2007	2008	2009	2010	2011	2012	2013	2014	Q3 2015
	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)
Operating System									
Android	-	691,378	7,011,633	71,101,570	243,422,810	501,424,674	802,274,495	1,060,470,064	844,264,871
BlackBerry OS	12,261,856	23,586,359	34,500,024	48,782,320	51,092,295	32,511,885	19,237,019	5,759,709	2,811,037
Firefox OS	-	-	-	-	-	-	521,681	1,290,988	743,992
iOS	3,706,982	13,725,911	25,108,519	47,522,874	93,137,769	135,865,176	153,448,185	192,650,301	156,750,979
Linux	14,044,374	17,570,423	10,530,476	9,159,479	14,459,348	14,399,459	7,143,530	4,256,791	1,954,744
Maemo/MeeGo	-	-	131,284	672,793	974,697	731,526	5,957	-	-
Other	18,840	-	-	-	-	-	-	-	-
Palm OS	2,111,395	2,656,107	439,773	18,067	100	-	-	-	-
RTOS	1,018,150,677	1,163,951,728	1,167,975,351	1,290,131,821	1,223,938,248	1,016,821,803	827,914,697	660,215,914	394,139,417
Sailfish OS	-	-	-	-	-	-	20,134	39,291	5,611
Symbian	78,685,797	72,976,958	77,925,146	110,760,290	81,540,530	23,886,192	2,564,653	2,288,238	1,332,000
Tizen	-	-	-	-	-	-	-	-	1,308,292
webOS	-	-	1,578,175	1,749,659	871,463	-	-	-	-
Windows Mobile	13,097,546	19,565,809	16,128,282	13,334,582	3,145,187	371,633	8,889	-	-
Windows Phone	-	-	-	1,579,496	5,802,765	17,512,281	33,513,424	34,907,034	24,256,336
Grand Total	**1,142,077,467**	**1,314,724,673**	**1,341,328,663**	**1,594,812,951**	**1,718,385,212**	**1,743,524,629**	**1,846,652,664**	**1,961,878,329**	**1,427,567,279**

U.S. Mobile Phone Shipments by Operating System

2007 - 2015

	2007	2008	2009	2010	2011	2012	2013	2014	Q3 2015
	(j)	(k)	(l)	(m)	(n)	(o)	(p)	(q)	(r)
Operating System									
Android	-	661,250	4,915,226	28,778,566	57,211,758	66,601,195	75,364,032	96,228,970	68,617,458
BlackBerry OS	8,256,238	15,560,203	21,446,438	19,705,279	10,037,346	2,514,384	1,882,046	579,071	365,509
Firefox OS	-	-	-	-	-	-	15,359	2,974	-
iOS	3,504,000	7,083,594	10,112,454	14,021,500	32,650,804	47,698,285	52,695,073	62,176,882	43,886,081
Linux	498,539	1,168,172	591,321	14,350	-	-	-	-	-
Maemo/MeeGo	-	-	3,340	2,500	-	-	-	-	-
Palm OS	1,818,656	2,443,732	347,006	6,078	-	-	-	-	-
RTOS	161,833,274	141,464,400	134,186,195	113,435,719	85,740,077	54,817,414	40,101,440	22,462,424	18,683,247
Symbian	509,564	613,923	1,159,543	2,036,542	765,095	363	-	-	-
webOS	-	-	1,485,513	1,607,466	829,650	-	-	-	-
Windows Mobile	5,507,451	6,422,887	5,744,644	2,953,027	304,320	-	-	-	-
Windows Phone	-	-	-	628,251	3,274,748	3,099,741	3,383,237	3,787,230	2,676,806
Grand Total	**181,927,722**	**175,418,161**	**179,991,680**	**183,189,278**	**190,813,798**	**174,731,382**	**173,441,187**	**185,237,551**	**134,229,101**

Note: Data is available from 2007 to Q3 2015.

Source: *IDC*, WW Quarterly Mobile Phone Tracker, November 13, 2015.

Appendix A

GREGORY K. LEONARD

Gregory K. Leonard is a Partner at Edgeworth Economics specializing in applied microeconomics and econometrics.

Dr. Leonard has written widely in the areas of antitrust, industrial organization, econometrics, intellectual property, class certification, and labor economics. His publications have appeared in the *RAND Journal of Economics*, the *Journal of Industrial Economics*, the *Journal of Econometrics*, the *International Journal of Industrial Organization*, the *Journal of Public Economics*, *Annales Economie et de Statistique*, the *Journal of Labor Economics*, the *International Journal of the Economics of Business*, *Antitrust Law Journal*, *Antitrust*, *Antitrust Source*, the *Journal of Economic Analysis & Policy*, *Journal of Competition Law and Economics*, the *Journal of Economic Surveys*, *法学家(Jurists' Review)*, *Antitrust Chronicle*, the *Berkeley Technology Law Journal*, the *Columbia Science and Technology Law Review*, the *European Competition Law Review*, *les Nouvelles*, *Landslide*, *Managing Intellectual Property*, *Legal Issues of Economic Integration*, *Kokusai Shoji Houmu (International Business Law and Practice)*, and the *George Mason Law Review*. Dr. Leonard authored two chapters and co-authored another chapter in the American Bar Association Section of Antitrust Law (ABA) volume *Econometrics* (2nd Ed., 2014), co-authored two chapters in the ABA volume *Issues in Competition Law and Policy*, and co-authored the "Econometrics and Regression Analysis" chapter of the ABA volume *Proving Antitrust Damages* (2nd Ed., 2010). He co-edited *Economic Approaches to Intellectual Property: Policy, Litigation, and Management* and authored or co-authored three of its chapters. One of these chapters (co-authored with Lauren J. Stiroh) was cited by the Court of Appeals for the Federal Circuit in its *Uniloc* decision. Dr. Leonard is a Senior Editor of the *Antitrust Law Journal* and has served as a referee for numerous economics journals.

Dr. Leonard was invited to speak on merger simulation at the 2004 US Department of Justice and Federal Trade Commission (FTC) Merger Workshop, the econometrics of evaluating competition in local retail markets at the 2008 FTC Retail Mergers Workshop, and the calculation of patent damages at the 2009 FTC Hearings on the Evolving IP Marketplace. The 2011 FTC report resulting from the latter hearings cited Dr. Leonard extensively. In 2005, Dr. Leonard served as a consultant on the issue of immunities and exemptions to the Antitrust Modernization Commission (AMC), which was tasked by Congress and the President with developing recommendations for changes to the US antitrust laws. He testified before the AMC in December 2005. Dr. Leonard gave an invited presentation on the use of natural experiments in antitrust at the European Commission's Directorate General for Competition (DG Comp) in 2014.

Dr. Leonard has extensive experience with international antitrust and intellectual property issues, particularly in Asia. He has been retained by the Anti-Monopoly Bureau of China's Ministry of Commerce (MOFCOM) as an outside economics expert to assist in merger reviews. Dr. Leonard has given invited presentations at MOFCOM, the Supreme People's Court of China, Renmin University, the Chinese Academy of Social Sciences, and the University of Political Science and Law. He was a member of ABA and US Chamber of Commerce delegations to joint workshops with the Chinese antitrust agencies, MOFCOM, NDRC, and SAIC, and served on the working groups of the ABA's Sections of Antitrust Law and International Law that prepared comments on MOFCOM's and SAIC's draft regulations. Dr. Leonard has also given presentations to the Japan Fair Trade Commission and the India Competition Commission.

Dr. Leonard has experience in a broad range of industries, including pharmaceuticals, telecommunications, airlines, semiconductors, hedge funds, securities, commercial and recreational fishing, medical devices, professional sports, credit card networks, payment systems, information services, computer software, computer hardware, chemicals, plastics, flat glass, retailing, advertising, beef processing, fertilizers, printing, petroleum, steel, beer, cereals, cosmetics, athletic apparel, film, milk, canned fish, vitamins, animal feed supplements, tissue, paperboard, industrial gas, concrete, automobiles, contact lens cleaners, sports beverages, soft drinks, diapers, tobacco products, graphite and carbon products, and modems, among others.

Dr. Leonard has provided written and oral testimony and presentations before federal and state courts, government agencies, and arbitration panels on issues involving antitrust, damages estimation, statistics and econometrics, surveys, valuation, and labor market discrimination.

Prior to joining Edgeworth, Dr. Leonard was a Senior Vice President at NERA and Lexecon Inc., a founding member and Director of Cambridge Economics, Inc., and an Assistant Professor at Columbia University, where he taught statistics, econometrics, and labor economics.

Dr. Leonard received an Sc.B. in Applied Mathematics-Economics from Brown University and a Ph.D. in Economics from the Massachusetts Institute of Technology, where he was a National Science Foundation Graduate Fellow and an Alfred P. Sloan Foundation Fellow.

EDUCATION

Massachusetts Institute of Technology

PhD, Economics, 1989
Alfred P. Sloan Foundation Fellowship, 1988-1989
National Science Foundation Graduate Fellowship, 1985-1988

Brown University

ScB, Applied Mathematics-Economics, 1985
Rohn Truell Memorial Premium in Applied Mathematics, 1985

PROFESSIONAL EXPERIENCE

2012-	Partner, Edgeworth Economics
2008-2012	Senior Vice President, NERA Economic Consulting
2004-2008	Vice President, NERA Economic Consulting
2000-2004	Senior Vice President, Lexecon, Inc.
1991-2000	Director, Cambridge Economics, Inc.

1990-1991 Senior Analyst, NERA Economic Consulting

1989-1990 Assistant Professor, Columbia University (Teaching Areas: Econometrics, Statistics, Labor Economics)

PAPERS AND PUBLICATIONS

"A Proposed Method for Measuring Competition Among Imperfect Substitutes," *Antitrust Law Journal* 60, 1992, pp. 889-900 (with J. Hausman and D. Zona).

"Issues in the Contingent Valuation of Environmental Goods: Methodologies for Data Collection and Analysis," in *Contingent Valuation: A Critical Assessment*, Ed. by J. A. Hausman, North Holland Press, 1993 (with D. McFadden).

"Assessing Use Value Losses Due to Natural Resource Injury," in *Contingent Valuation: A Critical Assessment*, ed. by J. A. Hausman, North Holland Press, 1993 (with J. Hausman and D. McFadden).

"Does Contingent Valuation Measure Preferences? Experimental Evidence," in *Contingent Valuation: A Critical Assessment*, ed. by J. A. Hausman, North Holland Press, 1993 (with P. Diamond, J. Hausman, and M. Denning).

"Competitive Analysis with Differentiated Products," *Annales d'Economie et de Statistique* 34, 1994, pp. 159-180 (with J. Hausman and D. Zona).

"A Utility Consistent, Combined Discrete Choice and Count Data Model: Assessing Recreational Use Losses Due to Natural Resource Damage," *Journal of Public Economics* 56, 1995, pp. 1-30 (with J. Hausman and D. McFadden).

"Market Definition Under Price Discrimination," *Antitrust Law Journal* 64, 1996, pp. 367-386 (with J. Hausman and C. Vellturo).

"Achieving Competition: Antitrust Policy and Consumer Welfare," *World Economic Affairs* 1, 1997, pp. 34-38 (with J. Hausman).

"Economic Analysis of Differentiated Products Mergers Using Real World Data," *George Mason Law Review* 5, 1997, pp. 321-346 (with J. Hausman).

"Superstars in the NBA: Economic Value and Policy," *Journal of Labor Economics* 15, 1997, pp. 586-624 (with J. Hausman).

"Efficiencies From the Consumer Viewpoint," *George Mason Law Review* 7, 1999, pp. 707-727 (with J. Hausman).

"Documents Versus Econometrics in Staples," contributed to www.antitrust.org, also available at www.nera.com (with J. Hausman).

"The Competitive Effects of a New Product Introduction: A Case Study," *Journal of Industrial Economics* 30, 2002, pp. 237-263 (with J. Hausman).

"Does Bell Company Entry into Long-Distance Telecommunications Benefit Consumers?" *Antitrust Law Journal* 70, 2002, pp. 463-484 (with J. Hausman and J. G. Sidak).

"On Nonexclusive Membership in Competing Joint Ventures," *RAND Journal of Economics* 34, 2003 (with J. Hausman and J. Tirole).

"Correcting the Bias When Damage Periods are Chosen to Coincide With Price Declines," *Columbia Business Law Review*, 2004, pp. 304-306 (with D. Carlton).

"Competitive Analysis Using a Flexible Demand Specification," *Journal of Competition Law and Economics* 1, 2005, pp. 279-301 (with J. Hausman).

"Using Merger Simulation Models: Testing the Underlying Assumptions," *International Journal of Industrial Organization* 23, 2005, pp. 693-698 (with J. Hausman).

"Application of Empirical Methods in Merger Analysis," report to the Fair Trade Commission of Japan, June 27, 2005 (with C. Dippon and L. Wu).

"A Practical Guide to Damages," in *Economic Approaches to Intellectual Property, Policy, Litigation and Management*, ed. by G. Leonard and L. Stiroh, 2005 (with L. Stiroh).

"Applying Merger Simulation Techniques to Estimate Lost Profits Damages in Intellectual Property Litigation," in *Economic Approaches to Intellectual Property, Policy, Litigation and Management*, ed. by G. Leonard and L. Stiroh, 2005.

"Antitrust Implications of Pharmaceutical Patent Litigation Settlements," in *Economic Approaches to Intellectual Property, Policy, Litigation and Management*, ed. by G. Leonard and L. Stiroh, 2005 (with R. Mortimer).

"Framework for Policymakers to Analyze Proposed and Existing Antitrust Immunities and Exemptions," report to the Antitrust Modernization Commission, October 24, 2005 (with D. Bush and S. Ross).

"Real Options and Patent Damages: The Legal Treatment of Non-Infringing Alternatives and Incentives to Innovate," *Journal of Economic Surveys* 20, 2006, pp. 493-512 (reprinted in *Economic and Legal Issues in Intellectual Property*, M. McAleer and L. Oxley, eds., Blackwell Publishing, 2007) (with J. Hausman).

"The Competitive Effects of Bundled Discounts," in *Economics of Antitrust: Complex Issues in a Dynamic Economy*, ed. by L. Wu, 2007.

"Estimation of Patent Licensing Value Using a Flexible Demand Specification," *Journal of Econometrics* 139, 2007, pp. 242-258 (with J. Hausman).

"Patent Damages and Real Options: How Judicial Characterization of Non-Infringing Alternatives Reduces Incentives to Innovate," *Berkeley Technology Law Journal* 22, Spring 2007, pp. 825-853 (with J. Hausman and J. G. Sidak).

"Don't Feed the Trolls," *les Nouvelles*, Vol. 42, September 2007, pp. 487-495 (reprinted in *Patent Trolls: Legal Implications*, C.S. Krishna, ed., The Icfai University Press, 2008) (with J. Johnson, C. Meyer, and K. Serwin).

"Are Three to Two Mergers in Markets with Entry Barriers Necessarily Problematic?" *European Competition Law Review* 28, October 2007, pp. 539-552 (with N. Attenborough and F. Jimenez).

"Economics and the Rigorous Analysis of Class Certification in Antitrust Cases," *Journal of Competition Law and Economics* 3, 2007, pp. 341-356 (with J. Johnson).

"Assessing the Competitive Effects of a Merger: Empirical Analysis of Price Differences Across Markets and Natural Experiments," *Antitrust*, Fall 2007, pp. 96-101 (with L. Wu).

"Incentives and China's New Antimonopoly Law," *Antitrust*, Spring 2008, pp. 73-77 (with F. Deng).

"Use of Simulation in Competitive Analysis," in *Issues in Competition Law and Policy*, ed. by W. Dale Collins, 2008 (with J.D. Zona).

"Allocative and Productive Efficiency," in *Issues in Competition Law and Policy*, ed. by W. Dale Collins, 2008 (with F. Deng).

"In the Eye of the Beholder: Price Structure as Junk Science in Antitrust Class Certification Proceedings," *Antitrust*, Summer 2008, pp. 108-112 (with J. Johnson).

"Merger Retrospective Studies: A Review," *Antitrust*, Fall 2008, pp. 34-41 (with G. Hunter and G. S. Olley).

"Roundtable Discussion: Developments—and Divergence—In Merger Enforcement," *Antitrust*, Fall 2008, pp. 9-27.

"Dispatch From China," *Antitrust*, Spring 2009, pp. 88-89.

"A Hard Landing in the Soft Drink Market – MOFCOM's Veto of the Coca-Cola/Huiyuan Deal," *Antitrust Chronicle*, April 2009(2) (with F. Deng and A. Emch).

"Predatory Pricing after *linkline* and *Wanadoo*," *Antitrust Chronicle,* May 2009(2) (with A. Emch).

"Farrell and Shapiro: The Sequel," *Antitrust*, Summer 2009, pp. 14-18 (with M. Lopez).

"掠夺性定价—美国与欧盟的法律及经济学分析" ("Predatory Pricing - Economics and Law in the United States and the European Union"), **法学家** (*Jurists' Review)*, 2009, pp. 100-110 (with A. Emch).

"Revising the Merger Guidelines: Second Request Screens and the Agencies' Empirical Approach to Competitive Effects," *Antitrust Chronicle,* December 2009(1) (with L. Wu).

"How Private Antitrust Litigation May Be Conducted in China," *Competition Law360*, January 6, 2010 (with F. Deng and W. Tang).

"Merger Screens: Market-Share Based Approaches and 'Upward Pricing Pressure,'" *Antitrust Source*, February 2010 (with E. Bailey, G. S. Olley, and L. Wu).

"Minimum Resale Price Maintenance: Some Empirical Evidence From Maryland," *BE Journal of Economic Analysis & Policy* 10, 2010 (with E. Bailey).

"Three Cases Reshaping Patent Licensing Practice," *Managing Intellectual Property*, March 2010 (with E. Bailey and A. Cox).

"Econometrics and Regression Analysis," in *Proving Antitrust Damages: Legal and Economic Issues*, ABA Section of Antitrust (2nd Edition), 2010 (with J. Langenfeld, W. Li, and J. Morris).

"Patent Damages: What Reforms Are Still Needed?," *Landslide* 2, May/June 2010 (with M. Lopez).

"The Google Books Settlement: Copyright, Rule 23, and DOJ Section 2 Enforcement," *Antitrust*, Summer 2010, pp. 26-31.

"The 2010 Merger Guidelines: Do We Need Them? Are They All We Need?," *Antitrust Chronicle*, October 2010(2).

"Evaluating the Unilateral Competitive Effects of Mergers Among Firms with High Profit Margins," *Antitrust*, Fall 2010, pp. 28-32 (with E. Bailey and L. Wu).

"Predatory Pricing in China—In Line With International Practice?," *Legal Issues of Economic Integration* 37, 2010, pp. 305-316 (with A. Emch).

"What Can Be Learned About the Competitive Effects of Mergers From 'Natural Experiments'?," *International Journal of the Economics of Business* 18, 2011, pp. 103-107 (with G. S. Olley).

"District Court Rejects the Google Books Settlement: A Missed Opportunity?," *Antitrust Source*, April 2011.

"Making Sense of 'Apportionment' in Patent Damages," *Columbia Science and Technology Law Review* 12, pp. 255-271, 2011 (with E. Bailey and M. Lopez).

"Rigorous Analysis of Class Certification Comes of Age," *Antitrust Law Journal* 77, 2011, pp. 569-586 (with J. Johnson).

"Economic Analysis in Indirect Purchaser Class Actions," *Antitrust*, Fall 2011, pp. 51-57 (with F. Deng and J. Johnson).

"Merger Assessment and Frontier of Economic Analyses (4): Empirical Methods in Antitrust Merger Review," *Kokusai Shoji Houmu (International Business Law and Practice)*, Vol. 40, No. 3, 2012, pp. 391-401 (with L. Wu)

"Merger Assessment and Frontier of Economic Analyses (5): Empirical Methods in Antitrust Merger Review," *Kokusai Shoji Houmu (International Business Law and Practice)*, Vol. 40, No. 4, 2012, pp. 557-564 (with L. Wu).

"Merger Assessment and Frontier of Economic Analyses (6): Empirical Methods in Antitrust Merger Review," *Kokusai Shoji Houmu (International Business Law and Practice)*, Vol. 40, No. 5, 2012, pp. 731-739 (with L. Wu).

"Economists' Roundtable on Hot Patent-Related Antitrust Issues", *Antitrust*, Summer 2013, pp. 10-21 (with D. Carlton, C. Meyer, C. Shapiro).

"Not So Natural Experiments," *Competition Policy International*, July 2013 (2).

"The Role of China's Unique Economic Characteristics in Antitrust Enforcement," in *China's Anti-Monopoly Law: The First Five Years*, ed. by Adrian Emch and David Stallibrass, 2013 (with F. Deng).

"Reflections on Bazaarvoice," *CPI Antitrust Chronicle*, March 2014 (1) (with P. Normann).

"An Introduction to Econometric Analysis," in *Econometrics: Legal, Practical and Technical Issues*, ABA Section of Antitrust (2nd Edition), 2014.

"The Econometric Framework," in *Econometrics: Legal, Practical and Technical Issues*, ABA Section of Antitrust (2nd Edition), 2014.

"Applying Econometrics to Estimate Damages," in *Econometrics: Legal, Practical and Technical Issues*, ABA Section of Antitrust (2nd Edition), 2014 (with J. Langenfeld, W. Li, and J. Morris).

"Determining RAND Royalties for Standard-Essential Patents," *Antitrust*, Fall 2014, pp. 86-94 (with M. Lopez).

"A Comparison of the Almost Ideal Demand System and Random Coefficients Logit Models For Use with Retail Scanner Data," NERA Working Paper, 2007 (with F. Deng).

PRESENTATIONS

"Merger Analysis with Differentiated Products," paper presented to the Economic Analysis Group of the US Department of Justice, April 1991 (with J. Hausman and D. Zona).

"Assessing Use Value Losses Due to Natural Resource Injury," paper presented at "Contingent Valuation: A Critical Assessment," Cambridge Economics Symposium, April 3, 1992 (with J. Hausman and D. McFadden).

"Contingent Valuation and the Value of Marketed Commodities," paper submitted to the Contingent Valuation Panel of the National Oceanic and Atmospheric Administration, U.S. Department of Commerce, August 12, 1992 (with J. Hausman).

"Economic Analysis of Differentiated Products Mergers Using Real World Data," paper presented to the George Mason University Law Review Antitrust Symposium, October 11, 1996 (with J. Hausman).

"Documents Versus Econometrics in Staples," paper presented to a program of the Economics Committee of the ABA Antitrust Section, September 5, 1997 (with J. Hausman).

Discussant, "New Developments in Antitrust" session, AEA meetings, January 7, 2000.

"In Defense of Merger Simulation," Department of Justice and Federal Trade Commission Merger Workshop, Unilateral Effects Session, February 18, 2004.

Discussant, "Proving Damages in Difficult Cases: Mock Trial & Discussion," NERA Antitrust & Trade Regulation Seminar, July 10, 2004.

"Network Effects, First Mover Advantage, and Merger Simulation in Damages Estimation," LSI Workshop on Calculating and Proving Patent Damages, July 16, 2004.

"Early Exchange of Documents," LSI Workshop on Pre- and Early Stage Patent Litigation, July 23, 2004.

"Lessons Learned From Problems With Expert Testimony: Antitrust Suits," LSI Workshop on Effective Financial Expert Testimony, November 4, 2004.

"Price Erosion and Convoyed Sales," LSI Workshop on Calculating & Proving Patent Damages, January 19, 2005.

"Economic Analysis of Rule 23(b)(3)," LSI Litigating Class Action Suits Conference, June 6, 2005.

"Early Exchange of Documents," LSI Workshop on Pre- & Early-Stage Patent Litigation, July 22, 2005.

"Issues to Consider in a Lost Profits Damages Analysis," Patent Litigation 2005, Practicing Law Institute, September 30, 2005.

"Antitrust Issues in Standard Setting and Patent Pools," Advanced Software Law and Practice Conference, November 3, 2005.

"New Technologies for Calculating Lost Profits," LSI Workshop on Calculating & Proving Patent Damages, February 27, 2006.

"Estimating Antitrust Damages," Fair Trade Commission of Japan, April 21, 2006.

"Economic Analysis of Rule 23(b)(3)," LSI Litigating Class Action Suits Conference, May 11, 2006.

"Permanent Injunction or Damages: What is the Right Remedy for Non-Producing Entities?," San Francisco Intellectual Property Law Association/Los Angeles Intellectual Property Law Association Spring Seminar, May 20, 2006.

"Antitrust Enforcement in the United States" and "Economic Analysis of Mergers," Sino-American Symposium on the Legislation and Practice of Anti-Trust Law, Beijing Bar Association, Beijing, People's Republic of China, July 17, 2006.

"Economic Analysis in Antitrust," Chinese Academy of Social Sciences, Beijing, People's Republic of China, July 20, 2006.

"Issues to Consider in a Lost Profits Damages Analysis," Patent Litigation 2006, Practicing Law Institute, September 26, 2006.

"Comparison of the Almost Ideal Demand System and Random Coefficient Models for Use With Retail Scanner Data," Pacific Rim Conference, Western Economic Association, Beijing, People's Republic of China, January 12, 2007 (with F. Deng).

Discussant, "Applied Economics" Session, Pacific Rim Conference, Western Economic Association, Beijing, People's Republic of China, January 12, 2007.

"Balancing IPR Protection and Economic Growth in China," International Conference on Globalization and the Protection of Intellectual Property Rights, Chinese University of Political Science and Law, Beijing, People's Republic of China, January 20, 2007.

"The Use and Abuse of Daubert Motions on Damages Experts: Lessons from Recent Cases," LSI Workshop on Calculating & Proving Patent Damages, February 27, 2007.

"Will Your Licenses Ever be the Same? Biotechnology IP Strategies," BayBio 2007 Conference, April 26, 2007.

"Tension Between Antitrust Law and IP Rights," Seminar on WTO Rules and China's Antimonopoly Legislation, Beijing, People's Republic of China, September 1, 2007.

"Issues to Consider in a Lost Profits Damages Analysis," Patent Litigation 2007, Practicing Law Institute, September 25, 2007.

Discussant, "Dominance and Abuse of Monopoly Power" Session, China's Competition Policy and Anti-Monopoly Law, J. Mirrlees Institute of Economic Policy Research, Beijing University, and the Research Center for Regulation and Competition, Chinese Academy of Social Sciences, Beijing, People's Republic of China, October 14, 2007.

"Opening Remarks," Seminar on China's Anti-monopoly Law and Regulation on Abuse of Intellectual Property Rights, Beijing, People's Republic of China, April 26, 2008.

"Issues to Consider in a Reasonable Royalty Damages Analysis," Patent Litigation 2008, Practicing Law Institute, October 7, 2008.

"Econometric Evaluation of Competition in Local Retail Markets," Federal Trade Commission and National Association of Attorneys General Retail Mergers Workshop, December 2, 2008

"Merger Review Best Practices: Competitive Effects Analysis," International Seminar on Anti-Monopoly Law: Procedure and Substantive Assessment in Merger Control, Beijing, People's Republic of China, December 15-17, 2008.

"The Use of Natural Experiments in Antitrust," Renmin University, Beijing, People's Republic of China, December 18, 2008.

"China's Antimonopoly Law: An Economist's Perspective," Bloomberg Anti-Monopoly Law of China Seminar, January 29, 2009.

Panelist, "Standards for Assessing Patent Damages and Their Implementation by Courts," FTC Hearings on the Evolving IP Marketplace, February 11, 2009.

"Economic Analysis of Agreements Between Competitors" and "Case Study: FTC Investigates Staples' Proposed Acquisition of Office Depot," Presentation to Delegation of Antitrust Officials from the People's Republic of China, Washington, DC, March 23, 2009.

"Reasonable Royalties in the Presence of Standards and Patent Pools," LSI Workshop, April 20, 2009.

Presentations on Unilateral Effects, Buyer Power, and the Intellectual Property-Antitrust Interface to Delegation from the Anti-Monopoly Bureau of MOFCOM of the People's Republic of China, Washington, DC, May 10-11, 2009.

Panelist, "The Use of Economic and Statistical Models in Civil and Criminal Litigation," Federal Bar Association, San Francisco, May 13, 2009.

"Trends in IP Rights Litigation and Economic Damages in China," Pursuing IP in the Pacific Rim, May 14, 2009.

Presentation on the Economics of Antitrust, National Judicial College of the People's Republic of China, Xi'an, People's Republic of China, May 25-26, 2009.

"Case Study: The Use of Economic Analysis in Merger Review," Presentation to the Anti-Monopoly Bureau of MOFCOM, Beijing, People's Republic of China, May 27, 2009.

"Economics and Antitrust Law," China University of Political Science and Law, Beijing, People's Republic of China, September 21, 2009.

"Case Study: Economic Analysis of Coordinated Interaction," Presentation to the Anti-Monopoly Bureau of MOFCOM, Beijing, People's Republic of China, September 22, 2009.

"Relevant Market Definition," 4th Duxes Antitrust Law Seminar, Beijing, People's Republic of China, September 26, 2009.

"Expert Economic Testimony in Antitrust Litigation," Supreme People's Court, Beijing, People's Republic of China, February 2, 2010.

"New Case Law for Patent Damages," Law Seminars International Telebriefing, April 28, 2010.

"China/India: Sailing in Unchartered Waters: Regulating Competition in the Emerging Economies – New Laws, New Enforcement Regimes and No Precedents," The Chicago Forum on International Antitrust Issues, Northwestern University School of Law Searle Center, May 20, 2010.

"Antitrust and Intellectual Property," Supreme People's Court, Beijing, People's Republic of China, May 26, 2010.

"Cartel Enforcement Trends in the United States," 2nd Ethical Beacon Anti-Monopoly Summit, Beijing, People's Republic of China, May 27, 2010.

Panelist, "The Future of Books and Digital Publishing: the Google Book Settlement and Beyond," 2010 American Bar Association Annual Meeting, August 7, 2010.

"Coordinated Effects" and "Non-Horizontal Mergers," Presentations to Delegation from India Competition Commission, US Chamber of Commerce, Washington, DC, October 26, 2010.

"UPP and Merger Simulation," Annual Conference of the Association of Competition Economics, Norwich, UK, November 11, 2010.

"Uniloc v. Microsoft: A Key Ruling For Patent Damages," Law Seminars International Telebriefing, January 21, 2011.

"Correlation, Regression, and Common Proof of Impact," New York City Bar Association, January 19, 2011.

"Private Litigation Under China's New Antimonopoly Law," Bar Association of San Francisco, February 17, 2011.

"Competition Law and State Regulation: Setting the Stage and Focus on State-Owned Enterprises," Competition Law and the State: International and Comparative Perspectives, Hong Kong, People's Republic of China, March 18, 2011.

Panelist, "Booking it in Cyberspace: The Google Book Settlement and the Aftermath," American Intellectual Property Law Association, San Francisco, May 13, 2011.

"Econometric Estimation of Cartel Overcharges," ZEW Conference on Economic Methods and Tools in Competition Law Enforcement, Mannheim, Germany, June 25, 2011.

Panelist, "Antitrust and IP in China," Antitrust and IP in Silicon Valley and Beyond, American Bar Association and Stanford University, Palo Alto, October 6, 2011.

Panelist, University of San Diego School of Law Patent Law Conference: The Future of Patent Law Remedies, January 18, 2013.

"Economics Framework," US-China Workshop on Competition Law and Policy for Internet Activities, China's State Administration for Industry and Commerce (SAIC) and the U.S. Trade and Development Agency (USTDA), Shenzhen, People's Republic of China, June 4-5, 2013.

Panelist, "China Inside and Out," American Bar Association, Beijing, People's Republic of China, September 16-17, 2013.

Panelist, "Remedies in Patent Cases," Fifth Annual Conference on The Role of the Courts in Patent Law & Policy, Berkeley and Georgetown Law Schools, November 1, 2013.

"Royalty Base," LeadershIP Conference, Qualcomm Incorporated, March 21, 2014.

"Reflections on Natural Experiments," DG Comp, April 8, 2014.

Panelist, "Antitrust in Asia: China," American Bar Association Section of Antitrust Law, Beijing, People's Republic of China, May 21-23, 2014.

Panelist, "Patent Damages Roundtable," 2015 Intellectual Property Institute, University of Southern California Gould School of Law, Los Angeles, March 23, 2015.

Panelist, "IP and Antitrust - The Current State of Economic Analysis," Global Competition Review Live 2nd Annual IP & Antitrust USA, Washington, DC, April 14, 2015.

Panelist, "FRAND Royalty Rates After Ericsson v. D-Link," American Bar Association, May 15, 2015.

PROFESSIONAL ACTIVITIES

Member, American Economic Association

Member, Econometric Society

Member, American Bar Association

Contributor, www.antitrust.org

Contributor, ABA Section of Antitrust Law, *Econometrics*, 2005

Associate Editor, *Antitrust*, 2007-2010

Senior Editor, *Antitrust Law Journal*, 2012-; Associate Editor, 2010-2012

Co-Editor, ABA Section of Antitrust Law Economics Committee Newsletter, 2009-2012

Member, Economics Task Force, ABA Section of Antitrust Law, 2011-2012

Member, ABA Delegation to International Seminar on Anti-Monopoly Law: Procedure and Substantive Assessment in Merger Control, Beijing, People's Republic of China, December 15-17, 2008

Member, Working Group for drafting the "Joint Comments of the American Bar Association Section of Antitrust Law and Section of International Law on the MOFCOM Draft Guidelines for Definition of Relevant Markets," 2009

Member, Working Group for drafting the "Joint Comments of the American Bar Association Section of Antitrust Law and Section of International Law on the SAIC Draft Regulations on the Prohibition of Acts of Monopoly Agreements and of Abuse of Dominant Market Position," 2009.

Member, Working Group for drafting the "Joint Comments of the American Bar Association Section of Antitrust Law and Section of International Law on the SAIC Draft Regulations on the Prohibition of Acts of Monopoly Agreements and of Abuse of Dominant Market Position," 2010.

Referee: *Econometrica*, *Review of Economics and Statistics*, *International Journal of Industrial Organization*, *Review of Industrial Organization, Journal of Sports Economics*, *Journal of Environmental Economics and Management*, *Research in Law and Economics*, *Labour Economics*, *Eastern Economic Journal*, *Journal of Forensic Economics, Antitrust, Antitrust Law Journal, Journal of Competition Law and Economics, Advances in Econometrics.*

TESTIMONY IN THE LAST FIVE YEARS

Edwards Lifesciences AG and Edwards Lifesciences, LLC v. CoreValve, Inc., United States District Court for the District of Delaware, C.A. No. 08-091 (GMS), 2009-2011 (Report, Deposition, Updated Report, Trial Testimony, Declarations).

WiAV Solutions, LLC v. Motorola, Inc., et al., United States District Court, Eastern District of Virginia, Richmond Division, Civil Action No. 3:09-cv-447-REP, 2010 (Report, Deposition).

In the Matter of CERTAIN NOTEBOOK COMPUTER PRODUCTS AND COMPONENTS THEREOF, before the United States International Trade Commission, Inv. No. 337-TA-705, 2010 (Report, Deposition).

Technology Patents, LLC v. Deutsche Telekom AG, et al., United States District Court, District of Maryland, Civil Action No. 8:07-cv-03012-AW, 2010 (Report).

Hollister Incorporated. v. C.R. Bard, Inc., United States District Court, Northern District of Illinois, Eastern Division, Civil Case No. 10-6427, 2011 (Deposition).

Quercus Trust v. LiveFuels, Inc., et al., Superior Court for the State of California, Civil No. 488685, 2011 (Deposition).

In re: Budeprion XL Marketing and Sales Practices Litigation, Civil Action 2:09-CV-2811, MDL Docket No. 2017, 2011 (Deposition).

Convolve, Inc. v. Dell Inc., et al., United States District Court, Eastern District of Texas, Marshall Division, Case No. No. 2:08-cv-244, 2011 (Deposition, Trial Testimony).

In the Matter of CERTAIN WIRELESS COMMUNICATION DEVICES, PORTABLE MUSIC AND DATA PROCESSING DEVICES, COMPUTERS AND COMPONENTS THEREOF, before the United States International Trade Commission, Investigation No. 337-TA-745, 2011 (Deposition).

In the Matter of CERTAIN MOBILE DEVICES, ASSOCIATED SOFTWARE, AND COMPONENTS THEREOF, before the United States International Trade Commission, Investigation No. 337-TA-744, 2011 (Deposition).

Oracle America, Inc. v. Google, Inc., United States District Court, Northern District for California, Case No. 3:10-CV-03561-WHA, 2011 (Deposition).

In the Matter of CERTAIN GAMING AND ENTERTAINMENT CONSOLES, RELATED SOFTWARE, AND COMPONENTS THEREOF, before the United States International Trade Commission, Investigation No. 337-TA-752, 2011 (Deposition).

General Atomics v. Paul Banks and TetraVue, Inc., Superior Court of the State of California, Case No. 37-2009-00084081-CU-BC-CTL, 2011 (Deposition, Trial Testimony).

Apple Inc., v. Motorola, Inc., United States District Court, Western District of Wisconsin, Case No. 10-CV-662 (BBC), 2011 (Deposition).

Genentech, Inc. and City of Hope v. Glaxo Group, Limited, et al., United States District Court, Central District of California, Western Division, Case No. 2:10-CV-02764-MRP (FMOx), 2011 (Deposition).

In the Matter of CERTAIN HANDHELD COMPUTING DEVICES, RELATED SOFTWARE, AND COMPONENTS THEREOF, before the United States International Trade Commission, Investigation No. 337-TA-769, 2011 (Deposition, Trial Testimony).

In the Matter of CERTAIN EQUIPMENT FOR COMMUNICATIONS NETWORKS, INCLUDING SWITCHES, ROUTERS, WIRELESS ACCESS POINTS, CABLE MODEMS, IP PHONES, AND PRODUCTS CONTAINING SAME, before the United States International Trade Commission, Investigation No. 337-TA-778, 2012 (Deposition).

Plantronics, Inc. v. Aliph, Inc., United States District Court for the Northern District of California, San Francisco Division, Case No. C09-01714 BZ, 2012 (Deposition).

Commonwealth Scientific and Industrial Research Organization v. Lenovo, Inc., et al., United States District Court for the Eastern District of Texas, Tyler Division, Case No. 6:09-cv-00400-LED, 2012 (Deposition).

Bayer HealthCare LLC v. Pfizer Inc., United States District Court, Northern District of Illinois, Eastern Division, Civil Action No. 1:12-cv-00630, 2012-2013 (Deposition).

L-7 Designs, Inc. v. Old Navy, Inc., United States District Court, Southern District of New York, Civil Action No. 09 Civ. 1432 (DC), 2012 (Deposition).

Apple, Inc. v. Motorola, Inc., United States District Court, Northern District of Illinois, Case No. 11-c-08540, 2012 (Deposition).

ITT Manufacturing Enterprises, Inc. v. Cellco Partnership, et al., United States District Court, District of Delaware, Civil Action No. 09-190-LPS, 2012 (Deposition).

Shelbyzyme LLC v. Genzyme Corporation, United States District Court, District of Delaware, Civil Action No. 09-768 (GMS), 2012 (Deposition, Trial Testimony).

In the Matter of CERTAIN DEVICES FOR IMPROVING UNIFORMITY USED IN A BACKLIGHT MODULE AND COMPONENTS THEREOF AND PRODUCTS CONTAINING THE SAME, before the United States International Trade Commission, Investigation No. 337-TA-805, 2012 (Deposition, Trial Testimony).

Rachel Eastman, et al. v. First Data Corporation, et al., United States District Court, District of New Jersey, Case No. 2:10-cv-04860 (WHW) (MCA), 2012 (Deposition).

In the Matter of CERTAIN COMMUNICATIONS EQUIPMENT COMPONENTS THEREOF, AND PRODUCTS CONTAINING THE SAME, INCLUDING POWER OVER ETHERNET TELEPHONES, SWITCHES, WIRELESS ACCESS POINTS, ROUTERS AND OTHER DEVICES USED IN LANs, AND CAMERAS, before the United States International Trade Commission, Investigation No. 337-TA-817, 2012 (Deposition).

Fujitsu Limited v. Belkin, et al., United States District Court, Northern District of California, San Jose Division, Case No. 10-cv-03972-LHK(PSG), 2012 (Deposition, Trial Testimony).

Medivation, Inc. v. The Regents of the University of California, et al., Superior Court of the State of California, Case No. CGC-11-510715, 2012 (Deposition, Trial Testimony).

In Re Photochromic Lens Antitrust Litigation (Direct Purchaser Action), United States District Court for the Middle District of Florida, Tampa Division, MDL Docket No. 2173, 2012 (Deposition, Hearing Testimony).

In Re Photochromic Lens Antitrust Litigation (Indirect Purchaser Actions), United States District Court for the Middle District of Florida, Tampa Division, MDL Docket No. 2173, 2012 (Deposition, Hearing Testimony).

In the Matter of CERTAIN PRODUCTS CONTAINING INTERACTIVE PROGRAM GUIDE AND PARENTAL CONTROL TECHNOLOGY, before the United States International Trade Commission, Investigation No. 337-TA-845, 2012 (Deposition, Trial Testimony).

In the Matter of CERTAIN COMPUTERS AND COMPUTER PERIPHERAL DEVICES AND COMPONENTS THEREOF AND PRODUCTS CONTAINING THE SAME, before the United States International Trade Commission, Investigation No. 337-TA-841, 2012-2013 (Trial Testimony).

Gemalto SA v. HTC Corporation, et al., United States District Court for the Eastern District of Texas, Tyler Division, Civil Action No. 6:10-CV-561-LED, 2013 (Deposition).

Adobe Systems Incorporated v. Wowza Media Systems, LLC, et al., United States District Court for the Northern District of California, Oakland Division, Case No. cv 11-02243, 2013 (Deposition).

In the Matter of CERTAIN AUDIOVISUAL COMPONENTS AND PRODUCTS CONTAINING THE SAME, before the United States International Trade Commission, Investigation No. 337-TA-837, 2013 (Deposition).

Ericsson Inc., et al. v. D-Link Corporation, et al., United States District Court for the Eastern District of Texas, Tyler Division, Civil Action No. 6:10-cv-473, 2013 (Deposition, Trial Testimony).

Edwards Lifesciences v. Medtronic CoreValve, et al., United States District Court for the District of Delaware, Case No. 12-23 (GMS), 2013 (Deposition, Trial Testimony).

Intellectual Ventures I LLC v. Trend Micro Incorporated and Trend Micro, Inc. (USA), United States District Court for the District of Delaware, C. A. No. 12-cv-1581-LPS, 2013 (Deposition).

The Money Suite Company v. Insurance Answer Center, LLC, et al., United States District Court for the Central District of California, Southern Division – Santa Ana, Lead Case No. 11-SACV-01847 AG (JPRx), 2013 (Deposition).

ParkerVision Inc. v. Qualcomm Incorporated, United States District Court for the Middle District of Florida, Jacksonville Division, Case No.: 3:11-cv-719-J-37-TEM, 2013 (Deposition, Trial Testimony).

Medtronic, Inc. v. Edwards Lifesciences Corporation, et al., United States District Court for the Central District of California, Case No.: SACV 12-00327 JVS (JPRx), 2013 (Deposition).

Microsoft Corporation v. Motorola Inc., et al., United States District Court for the Western District of Washington, Seattle Division, Case No. C10-1823JLR, 2013 (Deposition, Trial Testimony).

In the Matter of CERTAIN INTEGRATED CIRCUIT CHIPS AND PRODUCTS CONTAINING SAME, before the United States International Trade Commission, Investigation No. 337-TA-859, 2013 (Deposition, Trial Testimony).

Realtek Semiconductor Corporation v. LSI Corporation and Agere Systems, Inc., United States District Court Northern District of California, San Jose Division, Case No. 5:12-cv-03451 RMW, 2013 (Deposition, Trial Testimony).

Acer Inc., Acer America Corporation, and Gateway Inc. v. Technology Properties Limited, Patriot Scientific Corporation, and Alliacense Limited, United States District Court for the Northern District of California, San Jose Division, Case No. 5:08-cv-00877 PSG, 2013 (Deposition).

Intervet Inc. d/b/a Merck Animal Health, The Arizona Board of Regents on behalf of The University of Arizona v. Boehringer Ingelheim Vetmedica, Inc., United States District Court for the District of Delaware, Case No. 11-595-LPS, 2013 (Deposition).

In Re Innovatio IP Ventures, LLC Patent Litigation, United States District Court for the Northern District of Illinois, Case No. 1:11-cv-09308, 2013 (Deposition, Trial Testimony).

In the Matter of CERTAIN OMEGA-3 EXTRACTS FROM MARINE OR ACQUATIC BIOMASS AND PRODUCTS CONTAINING THE SAME, before the United States International Trade Commission, Investigation No. 337-TA-877, 2013 (Deposition).

Open Text SA v. Box Inc., United States District Court for the Eastern District of Virginia, Norfolk Division, Civil Action No. 2:13-CV-00319-MSD-DEM, 2013-2015 (Deposition, Trial Testimony).

Apple Inc. and Apple Sales International v. Motorola Mobility LLC, United States District Court for the Southern District of California, Case No. 3:12-cv-00355-GPC-BLM, 2013 (Deposition).

iControl Networks, Inc. v. Alarm.com Incorporated and Frontpoint Security Solutions, LLC, United States District Court for the Eastern District of Virginia, Alexandria Division, Case No. 1:13cv834 (LMB-IDD), 2013 (Deposition).

Affinity Labs of Texas, LLC v. General Motors LLC, United States District Court for the District of Eastern District of Texas, Beaumont Division, C.A. No. 1:12-cv-00582-RC, 2014 (Deposition).

W.L. Gore & Associates, Inc. v. C.R. Bard, Inc. and Bard Peripheral Vascular, Inc., United States District Court for the District of Delaware, C.A. No. 11-515-LPS-CJB, 2014 (Deposition).

Richard Noll and Rhythm Motor Sports, LLC v. eBay Inc., eBay Europe S.A.R.L., and eBay International AG, Inc., United States District Court for the Northern District of California, San Jose Division, Case No. 5:11-CV-04585-EJD, 2014 (Deposition).

Bristol-Myers Squibb Company v. Genentech Inc. and City of Hope, United States District Court for the Northern District of California, Western Division, Case No. 2:13-CV-05400-MRP (JEMx), 2014 (Report, Deposition).

Eli Lilly and Imclone v. Genentech Inc. and City of Hope, United States District Court for the Northern District of California, Western Division, Case No. 2:13-CV-07248-MRP, 2014 (Deposition).

Graftech International Ltd. and Graftech International Holdings Inc. F/K/A UCAR Carbon Company Inc. v. Carbone Savoie, Alcan France and Rio Tinto Alcan, International Chamber of Commerce, International Court of Arbitration, Case Ref.: 19798/AGF, 2014 (Report, Hearing Testimony).

Merit Medical Systems, Inc. v. Bard Access Systems, Inc., in the Third Judicial District Court for Salt Lake County, State of Utah, Civil No.: 130902435, 2014 (Report).

Samsung Electronics Co., Ltd. (Korea) v. Nokia Corporation (Finland), International Chamber of Commerce, International Court of Arbitration, Case Ref.: 19602/AGF/RD (c.19638/AGF), 2015 (Reports, Hearing Testimony).

Commonwealth Scientific and Industrial Research Organisation v. Mediatek Inc., et al., United States District Court for the Eastern District of Texas, Tyler Division, Case No. 6:12-cv-578 (LED), 2015 (Report, Rebuttal Report).

In the Matter of an Arbitration Under the Arbitration Act 1996 Between Teva Pharmaceutical Industries, Ltd. Teva Pharmaceuticals USA, Inc. and Starr Syndicate Limited, 2015 (Report).

In the Matter of CERTAIN NETWORK DEVICES, RELATED SOFTWARE, AND COMPONENTS THEREOF (I), before the United States International Trade Commission, Investigation No. 337-TA-944, 2015 (Report, Deposition).

Broadband iTV, Inc. v. Hawaiian Telecom, Inc., Oceanic Time Warner Cable, LLC, and Time Warner Cable, Inc., United States District Court for the District of Hawaii, Case No. 14-00169 ACK-RLP, 2015 (Declaration, Report, Deposition).

In the Matter of CERTAIN NETWORK DEVICES, RELATED SOFTWARE, AND COMPONENTS THEREOF (II), before the United States International Trade Commission, Investigation No. 337-TA-945, 2015 (Report, Deposition, Hearing Testimony).

SRI International, Inc. v. Cisco Systems, Inc., United States District Court for the District of Delaware, Case No. 13-1534 (SLR), 2015 (Report).

SELECTED MERGER EXPERIENCE

R.R. Donnelley/Meredith Burda (1990-1993): Merger of printing companies. Reviewed by the FTC. Preliminary Injunction Hearing. Part III Hearing.

Kimberly-Clark/Scott (1995): Merger of manufacturers of tissue products. Reviewed by the DOJ and the European Commission.

Staples/Office Depot (1996-1997): Proposed merger of office supply retailers. Reviewed by the FTC. Preliminary injunction hearing.

IMC/Western Ag (1997): Merger of mining companies. Reviewed by the DOJ.

Dow/Union Carbide (1999-2001): Merger of chemical manufacturers. Reviewed by the FTC.

Volvo/Scania (2000): Merger of truck manufacturers. Reviewed by the European Commission.

First Data/Concord (2003-2004): Merger of companies involved in merchant acquiring and payment networks. Reviewed by the DOJ.

Bumble Bee/Connors (2004): Merger of canned seafood manufacturers. Reviewed by the DOJ.

Sonaecom/Portugal Telecom (2006): Merger of telecommunications companies. Reviewed by the Portuguese Competition Authority.

Graphic Packaging/Altivity (2007-2008): Merger of paperboard manufacturers. Reviewed by the DOJ.

Inbev/Anheuser-Busch (2008): Merger of beer manufacturers. Reviewed by the DOJ, the UK Competition Commission, and MOFCOM.

Serta/Simmons (2009): Merger of mattress manufacturers. Reviewed by the FTC.

Coty/OPI (2010): Merger of nail polish manufacturers. Reviewed by the DOJ.

Knowles/NXP (2011): Knowles acquired the speaker/receiver business of NXP. Reviewed by MOFCOM.

AT&T/T-Mobile (2011): Consulted for the DOJ regarding the proposed deal between the two wireless service providers.

Confidential engagement for consumer product manufacturer (2012): Consulted for a consumer product manufacturer considering an acquisition with potential overlap in various jurisdictions around the world.

Confidential engagement for consumer product manufacturer (2012): Consulted for a consumer product manufacturer considering an acquisition with potential overlap in numerous product lines in the US.

UPS/TNT (2013): Consulted for the Ministry of Commerce of the People's Republic of China regarding the proposed deal between two package delivery services.

Thermo Fisher/Life Technologies (2014): Consulted for the Ministry of Commerce of the People's Republic of China regarding the proposed deal.

Appendix B

Appendix B
Documents Considered

Bates Documents

GOOG-00022380	GOOGLE-00395614	OAG00GLE0007622843
GOOG-00022381	GOOGLE-01-00024675	OAGOOGLE0000140295
GOOG-00022383	GOOGLE-01-00048436	OAGOOGLE0000140301
GOOG-00022386	GOOGLE-01-00049780	OAGOOGLE0000142142
GOOG-00022386	GOOGLE-01-00053552	OAGOOGLE0000287870
GOOG-00022388	GOOGLE-01-00064207	OAGOOGLE0000293785
GOOG-00103812	GOOGLE-01-00072883	OAGOOGLE0000478601
GOOG-00103813	GOOGLE-12-00080355	OAGOOGLE0000702509
GOOG-00103815	GOOGLE-12-00080356	OAGOOGLE0001342933
GOOG-00103816	GOOGLE-23-00000001	OAGOOGLE0003901182
GOOG-00130338	OAGOOGLE0013331514	OAGOOGLE0008258138
GOOG-00130343	OAGOOGLE0018885324	OAGOOGLE0009784791
GOOG-00132245	OAGOOGLE0018885325	OAGOOGLE0011726508
GOOG-00132625	OAGOOGLE002778855	OAGOOGLE0011761636
GOOG-00133931	OAGOOGLE0100003277	OAGOOGLE001208094
GOOG-00186879	OAGOOGLE0100005211	OAGOOGLE0012917834
GOOG-00186889	OAGOOGLE0100072599	OAGOOGLE0100166178
GOOGLE 01-00049780	OAGOOGLE0100072602	OAGOOGLE2000003711
GOOGLE-00-00000512	OAGOOGLE0100164986	OAGOOGLE2000077256
GOOGLE-00298438	OAGOOGLE2000181018	OAGOOGLE2000181018

Analyst Reports

"Android on Steroid: Google Enters Mobile Market with a Splash; Main Buy," Jefferies & Company, Inc., September 24, 2008.
"The Rise of Android, A Deep Dive Analysis," Caris & Company, September 13, 2010.
Bear Stearns, Apple Inc., "Raising Estimates and CY08 Target to $249 on Favorable Retail Survey and Asia Feedback".
Morgan Stanley, Apple Inc., "Shifting Focus to New Products and Margins".
UBS Investment Research, Apple Inc., "Macs Humming into Macworld," December 17, 2007.

Websites

"Google Reveal The Top 50 Android Applications (46 Public)," Chris Moor at Talk Android, http://www.talkandroid.com/92-developer-challenge-top-50-android-application/.

"101 Ways to Make Money with Your iPhone, Android or Mobile App," Monetize Pros, September 17, 2013, http://monetizepros.com/features/101-ways-to-make-money-with-your-iphone-android-or-mobile-app/.

"9 Things You Should Know About Firefox OS," Hongkiat Technology Design Inspiration, May 3, 2014, http://www.hongkiat.com/blog/9-things-about-firefox-os/.

"A Brief History of Android Phones," CNET, August 2, 2011
"About Display Network Only Campaigns," Google, https://support.google.com/adwords/answer/6340468
"Amazon Releases Kindle App for Android Phones," New York Times, June 28, 2010, http://bits.blogs.nytimes.com/2010/06/28/amazon-kindle-app-now-available-for-android/?pagemode=print.

"Android A to Z: What is a Kernel? Android Central, January 23, 2012, http://www.androidcentral.com/android-z-what-kernel.
"Android Development is 30% More Expensive Than iOS," Infinum, October 27, 2015, https://infinum.co/the-capsized-eight/articles/android-development-is-30-percent-more-expensive-than-ios.

"Android Interfaces and Architecture," Android, https://source.android.com/devices/index.html.
"Angry Birds" Now Available on Android for Free," Mashable, October 15, 2010, http://mashable.com/2010/10/15/angry-birds-android-2/#EmCWRAtkyqqF.

"App Makers Take Interest in Android," New York Times, October 25, 2010
"Apple's Biggest Breakthrough That Almost No One Knows About," Bloomberg, June 4, 2015.
"ART and Dalvik," Android, https://source.android.com/devices/tech/dalvik/.
"Average Cost to Develop a Mobile Phone App in 2012, By Operating System," Statista, 2012, http://www.statista.com/statistics/256541/average-cost-to-develop-an-app-by-os/.

"Becomes No. 1 iTunes Finance Application Within 24 Hours ," Mint, December 22, 2008, https://www.mint.com/press/mint-introduces-free-iphone-application.

"C is Number One!" ComputerWorld, April 7, 2010; "Google Exec Worries Over 'Rudderless' Java," PC World, April 14, 2010.

"C/C++ Programming I; Fundamental Programming Concepts, US San Diego Extension, http://extension.ucsd.edu/studyarea/index.cfm?vCourse=CSE-40475.

"C++ Programming, UC Berkeley Extension,

"Choosing a Membership," Apple Developer Website, https://developer.apple.com/support/compare-memberships/.

"Choosing the Campaign Type that's Right for You," Google, https://support.google.com/adwords/answer/2567043?hl=en

"Congratulations Google, Red Hat and the Java Community!" Jonathan's Blog, November 5, 2007, http://web.archive.org/web/20101023072550/http://blogs.sun.com/jonathan/entry/congratulations_google.

"Consumers Spend 85% of Time on Smartphones in Apps, But Only 5 Apps See Heavy Use," Tech Crunch, June 22, 2015,

"Create Your First App," Microsoft, https://msdn.microsoft.com/enus/library/windows/apps/bg124288.aspx

"Developing on Symbian, the World's Most Popular Mobile OS," DevX.com, July 22, 2010, http://www.devx.com/wireless/Article/45162

"Development Considerations," Android, http://developer.android.com/guide/topics/manifest/uses-sdk-element.html#considerations

"Dropbox for iPhones Is Out (and Awesome)," Mashable, September 29, 2009,

"Employer Costs for Employee Compensation: Historical Listing," Bureau of Labor Statistics, March 2004 - September 2015,

"Evernote for Android: It's Here!" Evernote, December 16, 2009, https://blog.evernote.com/blog/2009/12/16/evernote-for-android-its-here/

"Evernote News," Evernote, July 10, 2008, https://evernote.com/corp/news/pr/2008-07-10.php

"Facebook for iPhone Application Launches," Social Times, July 10, 2008,

"Facebook Launches Official Google Android Application," Mashable, September 8, 2009,

"Fifty Android Developers Get $25,000 Each: The List," TechCrunch, May 13, 2008,

"Financial Statements Glossary," Alphabet, https://abc.xyz/investor/other/additional-financial-information.html.

"Find Free Content & Preview Paid Content," Google, https://support.google.com/googleplay/answer/2851613?hl=en.

"First Google Android 2.0 Phone Arrives," CNET, October 30, 2009, http://www.cnet.com/news/first-google-android-2-0-phone-arrives/

"First Look – Samsung I7500 Preview," GSMArena, May 21, 2009, http://www.gsmarena.com/samsung_i7500-review-351.php

"First Look: LogMeIn Ignition," Engadget, January 19, 2009, http://www.engadget.com/2009/01/19/first-look-logmein-ignition/

"Five Reasons Why Google Android versus Apple iOS Market Share Numbers Don't Matter," Forbes, August 22, 2012,

"For Developers By Design," Ubuntu, http://www.ubuntu.com/phone/developers.

"Frequently Asked Questions About Google Financial Statements (through Q3 2015," Google,

"From Nexus One to Nexus 10; A Brief History of Google's Flagship Devices," ARS Technica, May 15, 2013,

"Fruit Ninja now available on Android!," Half Brick, September 17, 2010, http://halfbrick.com/fruit-ninja/fruit-ninja-on-android/

"Fruit Ninja," Gamespot, April 20, 2010, http://www.gamespot.com/fruit-ninja/

"Getting Started With Windows Mobile Application Development," Dr. Dobb's – The World of Software Development, September 22, 2006,

"GNU Classpath," http://www.gnu.org/software/classpath/license.html.

"Google Android 1.5," http://www.pcmag.com/article2/0,2817,2349058,00.asp

"Google Phone Update: Android 1.5 'Cupcake' Reviewed,"

"Google Phone Update: Android 1.5 'Cupcake' Reviewed", InformationWeek, June 2, 2009,

"Google Removes Applications Just Before Launch," Android Community,

"Google's Android Ambition is to Reshape the Mobile Industry, Report Says; But Android Faces Big Problems, Tight Deadlines, Says

"Google's Android Mobiles Overtake Global iPhone Sales," Financial Times, August 12, 2010

"Google's Mobile Ambitions; Launch of 'Android' Phone a Bold Bid for Dominance of Mobile Advertising," Richard Karpinski, BZMK,

"Happy Birthday to the Pandora App," Pandora Blog, July 10, 2013

"Homerun Battle 3D goes Goid, er… Droid," TouchMyApps, January 7, 2010,

"HomeRun Battle 3D," IGN, June 17, 2009, http://www.ign.com/games/home-run-battle-3d/iphone-14356201

"IcedTea: The First 100% Compliant Open-Source Java," InfoQ News, June 21, 2008, available at

"Inside Google's Android and Apple's iPhone OS as Core Platforms," Apple Insider, November 5, 2009

"Introducing Google Play: All Your Entertainment, Anywhere You Go," Google Official Blog,

"Java at 20 Years, Part 1: What's in a Name?" Application Development Trends Magazine, May 22, 2015.

"Java Platform, Standard Edition 8 Reference Implementations," Java.net, https://jdk8.java.net/java-se-8-ri/.

"Java Regains Spot as Most Popular Language in Developer Index," InfoWorld Tech Watch, April 14, 2015.

"Java WORA Defences Spent," ComputerWorld UK, May 30, 2012

"Java? It's So Nineties," Bloomberg Business, December 12, 2005.

"Java's Key to Success is Simplicity," JavaWorld, May 20, 2015.

"Kindle for Mac Now Finally Available," Engadget, March 18, 2010, http://www.engadget.com/2010/03/18/kindle-for-mac-now-finally-available/.

"Leading Java Software Vendors Endorse the SavaJe XE Operating System," PR Newswire, June 4, 2001.
"LogMeIn Ignition Launches Today, $29.99 in the Android Market," Phandroid, July 14, 2010, http://phandroid.com/2010/07/14/logmein-ignition-launches-today-29-99-in-the-android-market/

"Members," Open Handset Alliance, http://www.openhandsetalliance.com/oha_members.html. "Microsoft Pays Mobile App Developers to Catch Apple," Bloomberg, July 14, 2010, http://www.bloomberg.com/news/articles/2010-07-14/microsoft-pays-developers-to-build-mobile-apps-to-help-catch-up-with-apple

"Microsoft Still Paying Developers to Create Windows Phone Apps," CNET, April 6, 2012, http://www.cnet.com/news/microsoft-still-paying-developers-to-create-windows-phone-apps/

"Mint Comes to Android," Readwrite, May 3, 2010, http://readwrite.com/2010/05/03/mint_comes_to_android
"Motorola Droid (Verizon Wireless) review," CNET, October 28, 2009, http://www.cnet.com/products/motorola-droid-verizon-wireless/.

"Moving to OpenJDK as the official Java SE 7 Reference Implementation," Oracle blog, https://blogs.oracle.com/henrik/entry/moving_to_openjdk_as_the

"Nexus Phones Will Never See Huge Sales – But Here's Why They Don't Need To," Fortune, September 30, 2015, http://fortune.com/2015/09/30/google-nexus-smartphones-about-innovation-not-sales/.
"Once Declining Java Cements Its Lead in Language Popularity Index," InfoWorld Tech Watch, August 7, 2015
"Open Sourcing Is No Longer Optional, Not Even For Apple," Wired, June 9, 2015, http://www.wired.com/2015/06/open-sourcing-no-longer-optional-not-even-apple/.

"OpenJDK FAQ," Open JDK, December 18, 2010, available at http://openjdk.java.net/faq/.
"Open-source Java could result in port to iPhone; As the first anniversary of open-source Java approaches and Apple prepares to release an SDK for the iPhone, Sun thinks it may have a way to put Java on the popular handheld," InfoWorld Daily News, November 9, 2007.

"Our History in Depth," Google, https://www.google.com/about/company/history/.
"Pandora for Android," Pandora Blog, September 9, 2009, http://blog.pandora.com/2009/09/09/pandora_for_and/
"Publishing Your First App in the Play Store: What You Need to Know," Android Authority, May 20, 2014, http://www.androidauthority.com/publishing-first-app-play-store-need-know-383572/.

"Reservoir Devs: Why Apps Really Hit iOS Before Android," December 17, 2015, http://thenextweb.com/dd/2015/12/17/reservoir-devs-why-apps-really-hit-ios-before-android/#gref.

"Rovio's 'Angry Birds 2' to launch on Apple's iOS July 30," Apple Insider, July 16, 2015, http://appleinsider.com/articles/15/07/16/rovios-angry-birds-2-to-launch-on-apples-ios-july-30.

"Samsung Releases Bada SDK in Adavance of First Bada Phone," BetaNews, May 7, 2010, http://betanews.com/2010/05/07/samsung-releases-bada-sdk-in-advance-of-first-bada-phone/.

"Sing for Search Results with iPhone App," CNET, July 11, 2008, http://www.cnet.com/news/sing-for-search-results-with-iphone-app/.

"Smartphones: So Many Apps, So Much Time," Nielsen, July 1, 2014, http://www.nielsen.com/us/en/insights/news/2014/smartphones-so-many-apps--so-much-time.html.

"So Many Apps, So Much More Time for Entertainment," Nielsen, June 11, 2015, http://www.nielsen.com/us/en/insights/news/2015/so-many-apps-so-much-more-time-for-entertainment.html.

"Software Development for the Palm OS," Netmeister.org, 2000, https://www.netmeister.org/palm/PalmMisc/PalmMisc.html
"Sun Buys Up SavaJe, But Motive Remains Unclear," RCR Wireless News, April 16, 2007.
"Sun Fulfills Promise of Open and Free Java Technology and Releases Java SE Platform to OpenJDK Community; Advances OpenJDK Project with New Code, NetBeans Integration, Governance Board and Availability of Compatibility Tests," PR Newswire, May 8, 2007.

"Sun Starts Bidding Adieu to Mobile-Specific Java," CNET, October 19, 2007, http://news.cnet.com/8301-13580_3-9800679-39.html?part=rss&subj=news&tag=2547-1_3-0-20.

"Swift 2.0," Apple, June 8, 2015.
"The Best iPhone Apps of 2009," Tech Crunch, December 27, 2009, http://techcrunch.com/2009/12/27/best-iphone-apps-2009-appvee/.

"The Best of Android: Final Challenge Winners Announced," TechCrunch, August 28, 2008, http://techcrunch.com/2008/08/28/final-winners-of-android-challenge-announced/.

"The Dropbox Android App," Dropbox Blog, March 22, 2010, https://blogs.dropbox.com/dropbox/2010/03/the-dropbox-android-app-2/.

"The Fragmentation Effect," JavaWorld, May 24, 2004, http://www.javaworld.com/article/2072740/mobile-java/the-fragmentation-effect.html.

"The Future for Mobile Java," Eriks Diary, May 9, 2010, http://eriksdiary.blogspot.com (May 9, 2010).

"The RedMonk Programming Language Rankings: June 2015," RedMonk, June 2015, http://redmonk.com/sogrady/2015/07/01/language-rankings-6-15/

"The State of Mobile Apps," Nielsen, June 1, 2010, http://www.nielsen.com/us/en/insights/news/2010/the-state-of-mobile-apps.html.

"The Swift Effect: Apple's New Programming Language Means Way More iPhone Developers and Apps," The Verge, June 2, 2014, http://www.theverge.com/apple/2014/6/2/5773928/apple-swift-programming-developers-objective-c.

"The Weather Channel Tops 10 Million iPhone App Downloads," The Weather Company, April 28, 2010, http://www.theweathercompany.com/newsroom/2014/08/19/weather-channel-tops-10-million-iphone-app-downloads

"TIOBE Index for February 2016," TIOBE Software, February 2016, http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html.

"Top 10 Features You'll Love about Android 1.5," Geek, May 8, 2009, http://www.geek.com/android/top-10-features-youll-love-about-android-15-768061/

"Top 10 Features You'll Love about Android 1.5," http://www.geek.com/android/top-10-features-youll-love-about-android-15-768061/

"Top 30 Android Apps of All Time," TechCrunch, October 30, 2010, http://techcrunch.com/2010/10/30/top-30-android-apps/.

"Top Android Developers," Android APK.

"Top Ten Android Launch Apps," Tech Crunch, http://techcrunch.com/2008/10/22/top-ten-android-launch-apps/.

"Traffic Acquisition Cost – TAC," Investopedia, http://www.investopedia.com/terms/t/traffic-acquisition-cost-tac.asp.

"Under the Hood: Rebuilding Facebook for Android," Facebook, December 13, 2012, https://www.facebook.com/notes/facebook-engineering/under-the-hood-rebuilding-facebook-for-android/10151189598933920/

"Under the hood: Rebuilding Facebook for iOS," Facebook, August 23, 2012, https://www.facebook.com/notes/facebook-engineering/under-the-hood-rebuilding-facebook-for-ios/10151036091753920/.

"Why is Google Android Beating Symbian?" CNET, November 16, 2009, http://www.cnet.com/news/why-is-google-android-beating-symbian/ "Google Android Strategy."

"Write Once, Run Anywhere Not Working for Phones," CNET, July 15, 2005

"Zenonia and other Top Android Games of the Week," App Olicious, August 16, 2010, http://www.appolicious.com/articles/2794-zenonia-and-other-top-android-games-of-the-week

A Brief History of Android Phones, CNET, August 2, 2011, http://www.cnet.com/news/a-brief-history-of-android-phones/.

Alphabet (GOOG) Q3 2015 Results – Earnings Call Transcript, October 22, 2015, Porat at 2, http://seekingalpha.com/article/3596706-alphabet-goog-q3-2015-results-earnings-call-transcript.

Android, http://developer.android.com/guide/topics/renderscript/compute.html.

Developer Insights Report, IDC, August 2015.

Google Reveal The Top 50 Android Applications (46 Public), Chris Moor at Talk Android, http://www.talkandroid.com/92-developer-challenge-top-50-android-application/.

http://android-developers.blogspot.com/2008/10/android-market-now-available-for-users.html

http://arstechnica.com/apple/2013/02/apple-hq-also-targeted-by-hackers-will-release-tool-to-protect-customers/

http://arstechnica.com/security/2013/01/critical-java-vulnerabilies-confirmed-in-latest-version/

http://arstechnica.com/security/2013/01/critical-java-zero-day-bug-is-being-massively-exploited-in-the-wild/.

http://arstechnica.com/security/2013/01/javas-new-very-high-security-mode-cant-protect-you-from-malware/.

http://arstechnica.com/security/2013/01/massive-espionage-malware-relied-on-java-exploit-to-infect-pcs/.

http://arstechnica.com/security/2013/02/facebook-computers-compromised-by-zero-day-java-exploit/.

http://arstechnica.com/security/2013/02/microsoft-joins-apple-facebook-and-twitter-comes-out-as-hack-victim/.

http://arstechnica.com/security/2013/02/twitter-detects-and-shuts-down-password-data-hack-in-progress/.

http://blog.pandora.com/2013/07/10/happy-birthday-to-the-pandora-app/

http://extension.berkeley.edu/search/publicCourseSearchDetails.do?method=load&courseId=40931.

http://extension.ucsd.edu/studyarea/index.cfm?vCourse=CSE-40475.

http://finance.yahoo.com/q/hp?s=%5EGSPC&a=11&b=31&c=2007&d=11&e=31&f=2008&g=d&z=66&y=198.

http://www.artima.com/forums/flat.jsp?forum=106&thread=205707

http://www.cnet.com/news/homeland-security-still-advises-disabling-java-even-after-update/.

http://www.cnet.com/products/t-mobile-g1/
http://www.dataists.com/2010/12/ranking-the-popularity-of-programming-langauges/.
http://www.developer.nokia.com/Community/Discussion/
showthread.php?11133-j2me-compatibility-between-different-manufacuturers

http://www.infoworld.com/article/2617172/mac-os-x/apple-s-tim-cook-wins-where-steve-jobs-failed--on-java.html.
http://www.infragistics.com/community/blogs/nick-landry/archive/2013/
08/06/top-100-apps-availability-on-ios-android-windows-phone-amp-windows-8.aspx.

http://www.russellbeattie.com/blog/1005717
http://www.statista.com/statistics/256541/average-cost-to-develop-an-app-by-os/
http://www.theregister.co.uk/2012/08/30/oracle_knew_about_flaws/.
http://www.youtube.com/watch?v=9ei-rbULWoA&feature=results_main&playnext=1&list=PL10A2B0EBC7523D48
https://m.macnn.com/fullarticles/13/08/29/keeping.java.up.to.date.can.help.avoid.inconvenience/.
https://play.google.com/store?hl=en
https://redmonk.com/sograd y/category/programming-languages/
https://www.intego.com/mac-security-blog/apple-drops-java-in-latest-os-x-security-release/.
https://www.macnn.com/articles/13/08/29/keeping.java.up.to.date.can.help.avoid.inconvenience/.
https://www.quora.com/How-many-developers-companies-have-submitted-an-app-to-the
-iOS-App-Store-or-Android-Market/answer/Philipp-Berner?srid=oWse&share=1

Motorola 'Droid Does' Release: First Android 2.0 Phone Unveiled, Price Revealed (Updated, Photos), Huffington Post, May 25, 2011, http://www.huffingtonpost.com/2009/10/22/motorola-droid-does-relea_n_330100.html.

Occupational Employment and Wages: 15-1032 Computer Software Engineers, Systems Software, Bureau of Labor Statistics, May 2008, http://www.bls.gov/oes/2008/may/oes151032.htm.

Splash Play, AndroidTapp, http://www.androidtapp.com/splashplay/.
The Weather Channel App for Android Launches Major Updates," Lost Remote, February 28, 2013,
http://www.adweek.com/lostremote/the-weather-channel-app-for-android-launches-major-updates/37856

T-Mobile G1 Details, Price, and Launch Date Revealed, CNET, October 16, 2008,
http://www.cnet.com/news/t-mobile-g1-details-price-andlaunch-date-revealed/.

www.mono-project.com.
www.xamarin.com

Other Documents

M.J. Kim, Essays on the Economics of the Smartphone and Application Industry, University of Minnesota (2013)
"Additional Item: If Android Does Succeed It Will Be By Quite a Narrow Margin," Optical Networks Daily, August 7, 2009.
"An Exploratory Study of the Adoption of Mobile Development Platforms by Software Engineers," M. Muller, et al., 1st International Conference on Mobile Software Engineering and Systems, 2014.
"Android Tipped to Overtake iPhone by 2012," VNUNet United Kingdom, March 6, 2009
"App Annie Intelligence Product Suite Overview," Ann Annie.
"App Genome Report – February 2011," Lookout Mobile Security.
"comScore Mobile Metrix Methodology," comScore Inc., June 2013.
"comScore Reports April 2015 U.S. Smartphone Subscriber Market Share," comScore
"Global Smartphone Sales Forecast by Operating System and Region," Strategy Analytics, January 2011.
"Making Sense of a Fragmented World: Mobile Developer Economics 2010 and Beyond, Insights and Analysis from the Definitive Mobile Developer Survey Plus Benchmarks on the Platform Development Experience," VisionMobile Ltd, July 2010.

"Open Source Java Technology Debuts in GNU/Linux Distributions; Latest Releases of Fedora and Ubuntu Feature OpenJDK-Based Implementations," Business Wire, April 30, 2008.

"Platform Choice by Mobile App Developers," T. Bresnahan, et al., Stanford University, May 29, 2014 (henceforth "Bresnahan, et al. (2014)
"State of the Developer Nation Q1 2015," VisionMobile.
"Sun Announces Open Source Community Innovation Awards Program; Multi-Year Program Expected to Payout Millions to Foster Global Community-Based Innovation," Business Wire, December 5, 2007.

"Sun Fulfills Promise of Open and Free Java Technology and Releases Java SE Platform to OpenJDK Community; Advances OpenJDK Project with New Code, NetBeans Integration, Governance Board and Availability of Compatibility Tests;" PR Newswire, May 8, 2007.

"The Future of Java," Forrester, January 23, 2011. "The
State of Mobile", comScore, September 17, 2014

"U.S. Smartphone Use in 2015," Pew Research Center, April 1, 2015
17 U.S.C. § 504(b)
Android Activations, June 2015.
AppAnnie Daily DNA Data, App Annie, January 26, 2016.
AppAnnie data.
Applications, Androlib.com.
Chow, et al., Factors Affecting the Demand of Smartphone Among Young Adult, Multimedia University (2012)
comScore data.
Conversation with Jonathan Gold.
Federal Trade Commission, The Evolving IP Marketplace: Aligning Patent Notice and Remedies with Competition, March 2011.
Fourteenth Annual Report and Analysis of Competitive Market Conditions With Respect to Mobile Wireless, Including Commercial Mobile Services, Federal Communications Commission, May 20, 2010.
Google Inc. 10-K, 2005 Annual Report, December 31, 2005.
Google Inc. 10-K, 2006 Annual Report, December 31, 2006.
Google Inc. 10-K, 2007 Annual Report, December 31, 2007.
Google Inc. 2008 10-K, December 31, 2008.
Google Inc. 2009 10-K, December 31, 2009.
Google Inc. 2010 10-K, December 31, 2010.
Google Inc. 2011 10-K, December 31, 2011.
Google Inc. 2012 10-K, December 31, 2012.
Google Inc. 2013 10-K, December 31, 2013.
Google Inc. 2014 10-K, December 31, 2014.
Google Inc. 2015 3Q 10-Q, September 30, 2015.
Gross National Product: Implicit Price Deflator, U.S. Bureau of Economic Analysis, updated December 22, 2015.
Interview of Anwar Ghuloum.
Interview of John Rizzo.
Interviews of Tim Bray, Brian Swetland.
ITG Monthly Mobile Handset Report, ITG, December 2015
J. Pearl, Causality: Models, Reasoning, and Inference (2nd Ed., 2009)
J.C. Rochet and J. Tirole, "Platform competition in two-sided markets," Journal of the European Economic Association (2003)
J.C. Rochet and J. Tirole, "Two-sided markets: a progress report," RAND Journal of Economics (2006)
Larry Ellison Trial Transcript, Volume 2, April 17, 2012
M. Bohmer, et al., "Falling Asleep with Angry Birds, Facebook and Kindle – A Large Scale Study on Mobile Application Usage," MobileHCI 2011, September 2, 2011.
M.J. Kim, Essays on the Economics of the Smartphone and Application Industry, University of Minnesota, Ph.D. Thesis (2013).
Market Data,App Annie, data as of December 18, 2015.
Mobile Metrix, comScore, Inc., January 19, 2016.
Number of Android Applications, AppBrain.com.
Polar Bear Prods., Inc. v. Timex Corp., 384 F.3d (9th Cir. 2004), No. 03-35188, 2004 WL 2376507.
Polar Bear Prods., Inc. v. Timex Corp., 384 F.3d 700, 711 (9th Cir. 2004), as amended on denial of reh'g and reh'g en banc (Oct. 25, 2004), opinion amended on denial of reh'g, No. 03-35188, 2004 WL 2376507 (Oct. 25, 2004).
R. Brealey and S. Myers, Principles of Corporate Finance (5th Ed., 1996)
S. Berry, Estimating Discrete-Choice Models of Product Differentiation, Econometrica (1994)
Simonson Report.
Sixteenth Annual Report and Analysis of Competitive Market Conditions With Respect to Mobile Wireless, Including Commercial Mobile Services, Federal Communications Commission, March 21, 2013.
Sun Microsystems, 2009 Form 10-K, filed August 28, 2009.
T. Bresnahan, J. Davis, and P.L. Yin, "Economic Value Creation in Mobile Applications," in The Changing Frontier: Rethinking Science and Innovation Policy (A. Jaffe and B. Jones, eds., 2015) (hereafter "Bresnahan, et al. (2015)), at location 5838 (Kindle version).
Uniloc USA, Inc. v. Microsoft Corp., Nos. 2010-1035, 2010-1055, 2011 WL 9738 (Fed. Cir. Jan. 4, 2011).
Weingaertner Decl. at ¶¶ 24-29 (Google Daubert Motion Exs. X-CC)
WW Quarterly Mobile Phone Tracker - 2015Q3 Historical Release, IDC, 2015
WW Quarterly Mobile Phone Tracker, IDC, November 13, 2015.
Y. Liu, "Mobile App Platform Choice: An Application of Strategic Games on Big Data," November 2014.

Expert Report of Dr. Alan J. Cox, October 3, 2011.
Expert Report of Dr. Chris F. Kemerer, January 8, 2016.
Expert Report of Dr. Cockburn, February 3, 2012 and revised February 9, 2012.
Expert Report of Dr. Leonard, February 8, 2016.
Expert Report of Robert Zeidman, January 8, 2016.
Expert Reports of James E. Malackowski, January 8, 2016 and corrected.
Opening Expert Report of Dr. Owen Astrachan, July 29, 2011
Rebuttal Expert Report of Dr. Owen Astrachan, August 12, 2011.

Transcript and Exhibits of Jury Trial Proceedings: Oracle America, Inc. vs Google, Inc., Volume 1.
Transcript and Exhibits of Jury Trial Proceedings: Oracle America, Inc. vs Google, Inc., Volume 2.
Transcript and Exhibits of Jury Trial Proceedings: Oracle America, Inc. vs Google, Inc., Volume 3.
Transcript and Exhibits of Jury Trial Proceedings: Oracle America, Inc. vs Google, Inc., Volume 4.
Transcript and Exhibits of Jury Trial Proceedings: Oracle America, Inc. vs Google, Inc., Volume 5.
Transcript and Exhibits of Jury Trial Proceedings: Oracle America, Inc. vs Google, Inc., Volume 6.
Transcript and Exhibits of Jury Trial Proceedings: Oracle America, Inc. vs Google, Inc., Volume 7.
Transcript and Exhibits of Jury Trial Proceedings: Oracle America, Inc. vs Google, Inc., Volume 8.
Transcript and Exhibits of Jury Trial Proceedings: Oracle America, Inc. vs Google, Inc., Volume 9.
Transcript and Exhibits of Jury Trial Proceedings: Oracle America, Inc. vs Google, Inc., Volume 10.
Transcript and Exhibits of Jury Trial Proceedings: Oracle America, Inc. vs Google, Inc., Volume 11.
Transcript and Exhibits of Jury Trial Proceedings: Oracle America, Inc. vs Google, Inc., Volume 12.
Transcript and Exhibits of Jury Trial Proceedings: Oracle America, Inc. vs Google, Inc., Volume 13.
Transcript and Exhibits of Jury Trial Proceedings: Oracle America, Inc. vs Google, Inc., Volume 14.
Transcript and Exhibits of Jury Trial Proceedings: Oracle America, Inc. vs Google, Inc., Volume 15.
Transcript and Exhibits of Jury Trial Proceedings: Oracle America, Inc. vs Google, Inc., Volume 16.

Deposition of Aditya Kumar Agarwal, April 8, 2011.
Deposition of Alan Brenner, Volume I, December 15, 2015.
Deposition of Alan J. Cox, October 26, 2011.
Deposition of Andrew Mcfadden, May 4, 2011.
Deposition of Anwar Ghuloum, December 9, 2015.
Deposition of Bob Lee, August 3, 2011.
Deposition of Brian J. Swetland, July 7, 2011.
Deposition of Craig Gering, July 20, 2011.
Deposition of Daniel Morrill, July 12, 2011.
Deposition of David K. Hofert, December 1, 2015.
Deposition of David Mazieres, September 8, 2011.
Deposition of Dennis Allison, September 9, 2011.
Deposition of Donald Smith, November 20, 2015.
Deposition of Douglas Kehring, July 28, 2011.
Deposition of Edward Screven, July 29, 2011.
Deposition of Edward Senteno, November 18, 2015.
Deposition of Erez Landau, September 14, 2011.
Deposition of Eric Chu, April 28, 2011.
Deposition of Eric E. Schmidt, August 23, 2011.
Deposition of Geoffrey A. Morton, June 7, 2011.
Deposition of Georges Saab, Volume I, December 16, 2015.
Deposition of Hasan Rizvi, July 28, 2011.
Deposition of Henrik Stahl, January 14, 2016.
Deposition of Hiroshi Lockheimer, December 8, 2015.
Deposition of Jack W. Davidson, September 9, 2011.
Deposition of James Kearl, March 26, 2012.
Deposition of James Kolotouros, Volume I, January 26, 2016.
Deposition of Jeet Kaul, August 5, 2011.
Deposition of John Pampuch, July 29, 2011.
Deposition of Jonathan Schwartz, July 20, 2011.
Deposition of Joshua Bloch, July 8, 2011.
Deposition of Larry Page, August 24, 2011.
Deposition of Lawrence J. Ellison, August 12, 2011.
Deposition of Leo Cizek, July 22, 2011.
Deposition of Lisa J. Ripley, April 14, 2011. Deposition of Mark B. Reinhold, August 5, 2011. Deposition of Mark Wayne, Volume I, December 3, 2015. Deposition of Mike Ringhofer, December 2, 2015. Deposition of Nedim Fresko, May 10, 2011.
Deposition of Noel Poore, September 7, 2011.

Deposition of Owen Astrachan, September 9, 2011.
Deposition of Param V. Singh, June 23, 2011.
Deposition of Patrick Brady, July 21, 2011.
Deposition of Peter B. Kessler, August 4, 2011.
Deposition of Peter Edward Lord, July 22, 2011.
Deposition of Rachel A. Claflin, April 26, 2011.
Deposition of Rafael Camargo, September 8, 2011.
Deposition of Reto Meier, Volume I, December 11, 2015.
Deposition of Richard Miner, May 26, 2011.
Deposition OF Robert B.K. Dewar, September 13, 2011.
Deposition of Robert Griesemer, June 23, 2011.
Deposition of Robert Vandette, September 7, 2011.
Deposition of Steven G. Harris, July 28, 2011.
Deposition of Steven M. Shugan, September 26, 2011.
Deposition of Susan D. Wojcicki, August 31, 2011.
Deposition of Terence Parr, September 15, 2011.
Deposition of Terrence Barr, Volume I, December 9, 2015.
Deposition of Tim Lindholm, September 7, 2011.
Deposition of Urs Holzle, Volume I, November 24, 2015.
Deposition of Vineet Gupta, July 26, 2011.
Deposition of William S. Rutledge III, Volume I, December 9, 2015
Depositions of Andrew E. Rubin, April 5, July 7, July 27 and August 18, 2011.
Depositions of Daniel Bornstein, May 16, July 22, and November 21, 2011.
Depositions of Felix Lin, Volume I and II, December 14 and 18, 2015.
Depositions of John C. Mitchell, September 2, 6 and 7, 2011.
Depositions of Jon Gold, Volume I and II, December 11, 2015 and January 29, 2016.

GOOG-00022382
OAGOOGLE0013561757
OAGOOGLE2000462635
Developer Insights Report, IDC, August 2015
Astrachan Opening Report , January 8, 2016
Google Analysis of Top Applications Programming Languages
"Apple's App Store: 15,000 apps. Google's Android Marketplace: 800 apps," Venture Beat, January 22, 2009, http://venturebeat.com/2009/01/22/apples-app-store-15000-apps-googles-android-marketplace-800-apps/
"Stats: Android Market is No App Store Killer, But Not a Dud Either," Mashable, October 31, 2008, http://mashable.com/2008/10/31/android-market-stats/#mtaev1Srzaq3
"State of the Apps — iPhone: 10,000 Apps, 300 million Downloads. Android: 462 Apps," Venture Beat, December 5, 2008, http://venturebeat.com/2008/12/05/state-of-the-apps-iphone-10000-apps-300-million-downloads-android-462-apps/